

## Durham Research Online

---

### Deposited in DRO:

22 August 2018

### Version of attached file:

Published Version

### Peer-review status of attached file:

Peer-reviewed

### Citation for published item:

Farrow, Joseph A. (2018) 'A Monte Carlo approach to the 4D scattering equations.', Journal of high energy physics., 2018 (8). 085.

### Further information on publisher's website:

[https://doi.org/10.1007/JHEP08\(2018\)085](https://doi.org/10.1007/JHEP08(2018)085)

### Publisher's copyright statement:

This article is distributed under the terms of the Creative Commons Attribution License (CC-BY 4.0), which permits any use, distribution and reproduction in any medium, provided the original author(s) and source are credited.

### Additional information:

## Use policy

---

The full-text may be used and/or reproduced, and given to third parties in any format or medium, without prior permission or charge, for personal research or study, educational, or not-for-profit purposes provided that:

- a full bibliographic reference is made to the original source
- a [link](#) is made to the metadata record in DRO
- the full-text is not changed in any way

The full-text must not be sold in any format or medium without the formal permission of the copyright holders.

Please consult the [full DRO policy](#) for further details.

# A Monte Carlo approach to the 4D scattering equations

**Joseph A. Farrow**

*Department of Mathematical Sciences, Durham University,  
Durham, DH1 3LE, United Kingdom*

*E-mail:* [joseph.a.farrow@durham.ac.uk](mailto:joseph.a.farrow@durham.ac.uk)

**ABSTRACT:** The scattering equation formalism is a general framework for calculation of amplitudes in theories of massless particles. We provide a detailed introduction to the 4D scattering equation framework accessible to non-experts, outline current difficulties solving the equations numerically, and explain how to overcome them with a Monte Carlo algorithm. With this submission we include `treeamps4dJAF`, the first publicly available MATHEMATICA package for calculating amplitudes by solving the scattering equations, supporting MHV analytical and  $N^{k-2}$ MHV numerical computations. The package provides a powerful and flexible computational tool for calculating tree-level amplitudes in super Yang Mills theories, Einstein supergravity and conformal supergravity. We tabulate sets of numerical solutions up to 9 points in all MHV sectors and 12 points in the NHMV sector which can be used for fast evaluation of amplitudes.

**KEYWORDS:** Scattering Amplitudes, Supergravity Models, Supersymmetric Gauge Theory

**ARXIV EPRINT:** [1806.02732](https://arxiv.org/abs/1806.02732)

---

## Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Review and conventions</b>	<b>3</b>
<b>3</b>	<b>Calculating amplitudes with the 4D scattering equations</b>	<b>6</b>
<b>4</b>	<b>Numerical methods</b>	<b>9</b>
4.1	Difficulties with CHY's inverse soft algorithm	10
4.2	Monte Carlo algorithm	12
4.3	Efficient component amplitude extraction	16
<b>5</b>	<b>Conclusions</b>	<b>16</b>
<b>A</b>	<b>The MATHEMATICA package treeamps4dJAF</b>	<b>18</b>
A.1	Calculating amplitudes	18
A.2	Solving the 4D scattering equations	20
A.3	Applying momenta and solutions to expressions	21
A.4	Grassmann integration and swapping left set algorithms	22
A.5	File input-output for sets of solutions	23
<b>B</b>	<b>Analytical details of the 4D scattering equations</b>	<b>23</b>
B.1	Recovering the general $d$ scattering equations for $d = 4$	23
B.2	Symmetries, little group scaling and Grassmanians	25
B.3	Deleting equations and momentum conservation	27
B.4	Permutations and choice of left set	28
B.5	The Jacobian of the scattering equations	29

---

## 1 Introduction

The calculation of scattering amplitudes in quantum field theories is both an essential ingredient in interpreting data from high energy physics experiment, and a powerful tool for probing the deepest questions of theoretical physics. The development of new mathematical techniques for calculating amplitudes has lead to important advances in both of these areas. In the 1980s the introduction of spinor helicity notation gave way to new simplified computations of four dimensional amplitudes which previously had seemed intractable, for example [1–3]. In recent years there have been many advances in the set of technical tools that exist for calculating perturbative scattering amplitudes. Key tree-level techniques include recursive methods for calculation of higher point amplitudes from lower point inputs, known as BCFW recursion [4, 5], and the formulation of field theory amplitudes in terms of string worldsheet calculations, known as twistor-string theory [6–8].

The simplicity of the Parke-Taylor form of the Yang-Mills MHV amplitude [3] inspired a description in terms of 2D current algebra [9]. This idea was then generalised to  $\mathcal{N} = 4$  super Yang-Mills amplitudes [6–8], and was also found to calculate  $\mathcal{N} = 4$  conformal supergravity amplitudes [10]. A similar worldsheet formula for  $\mathcal{N} = 8$  supergravity was found in [11]. The spectra of these string models contains only field theory degrees of freedom. Cachazo, He and Yuan extended the formula of [11] to a framework for calculating scattering of particles in arbitrary dimensions for a wide variety of theories in terms of a unified set of scattering equations [12–14]. We will refer to these equations as ‘general  $d$  scattering equations’. In the CHY framework, tree-level amplitudes for different theories of massless particles are supported on the solutions of these scattering equations, which were first discovered in the context of ordinary string theory in [15, 16]. Worldsheet expressions of this form are related in a deep way to recursive formulae arising from BCFW [17–20].

The power of the spinor helicity formalism in four dimensions along with the generality of the scattering equation formalism were then combined to produce 4D ambitwistor string theory [21, 22]. This model gives a worldsheet description for the tree-level S-matrices of Yang-Mills theories and Einstein gravity which are valid for any number of supersymmetries and are supported on refined scattering equations which are graded by helicity degree. We will refer to these equations as ‘4D scattering equations’. In recent work we extended this formalism to include  $\mathcal{N} = 4$  conformal supergravity [23].

The scattering equations are understood in most detail at tree level, and this paper will consider only tree level amplitudes. There are extensions to the general  $d$  scattering equations to calculate loop integrands [24–27]. The 4D scattering equations currently only support tree level computations, although some preliminary one loop expressions exist [18].

Formulae supported on scattering equations have been successful in representing and calculating abstract theoretical properties of amplitudes such as soft limits [28–32], collinear limits [33] and relations between the S-matrices of different theories [14, 34]. Direct evaluation of amplitudes by solving the scattering equations is difficult, and some approaches solve the integrations by different methods [35, 36]. The equations have  $(n - 3)!$  solutions at  $n$  points, and it is likely that finding all of these solutions analytically for generic kinematics is not possible. Calculating amplitudes in this framework then becomes a primarily numerical problem, which has been addressed only for the general  $d$  equations. These equations can be reduced to a simplified polynomial form [37], which MATHEMATICA’s in-built NSolve algorithm can solve numerically up to 9 points on a standard laptop. CHY provide an algorithm for finding individual solutions at higher points [12], but there are difficulties finding all solutions in this way. Solutions which start out distinct at the start of the algorithm generically degenerate and the full set of solutions is not recovered at the end; details are discussed in this paper. The 4D equations do not currently have an equivalent simplified polynomial form and depend on a larger set of variables than the general  $d$  equations, and as such NSolve cannot solve them above 7 points. We overcome these difficulties with a Monte-Carlo solution finding algorithm which finds all solutions.

There exist published MATHEMATICA packages for evaluation of tree level amplitudes using BCFW [38, 39], but to date there have been no equivalent packages for calculating amplitudes using the scattering equations either analytically or numerically. [40] provide an

algorithm for calculation of amplitudes numerically without directly solving the scattering equations, but no explicit implementation is given. Available MATHEMATICA packages focus on Yang-Mills theory, and we provide the first explicit publicly available algorithms for calculating Einstein supergravity and  $\mathcal{N} = 4$  conformal supergravity amplitudes at tree level.

The key purpose of this paper is to present the MATHEMATICA package `treeamps4dJAF`, which explicitly calculates tree level amplitudes by solving the scattering equations. The analytical framework required for understanding the details of the 4D scattering equation formalism is covered, and we provide details of how to find full sets of solutions for a given set of numerical momenta and MHV degree by Monte Carlo algorithm. The package supports analytical computations in the MHV sector, and numerical computations in general MHV sectors for a wide variety of theories.

We hope that this paper, along with its associated package, will provide a computational tool that can serve for a number of different purposes. It can be seen as an lead-in to the scattering equation formalism those new to the field, as well as a general introduction to calculating amplitudes in MATHEMATICA. It can be used as a reference to check if new techniques for calculating amplitudes agree with previously known results for the theories currently supported, and a play box for discovering further theories that can be described in the 4D scattering equation formalism. We intend to update the package with integrands for new theories once they have been discovered.

The structure of the paper is as follows. We start with a review of the general  $d$  and 4D scattering equations and the integrands for currently supported theories in section 2. Section 3 gives an overview of the analytical tools necessary to calculate amplitudes from the 4D scattering equation integral. Numerical methods for solving the equations outside of the MHV sector by Monte Carlo algorithm and for extracting component amplitudes from a superamplitude are detailed in section 4. We conclude and discuss further directions in section 5. Appendix A introduces the MATHEMATICA package `treeamps4dJAF` which gives a concrete implementation of all of the algorithms discussed in the paper and is included with the paper. The package's key functions are detailed in this appendix along with usage examples. Finally appendix B gives detailed proofs of various properties of the 4D scattering equations, including those used in section 3.

## 2 Review and conventions

In this section we review relevant details of the general  $d$  and 4D scattering equation formalism. For an  $n$  point  $N^{k-2}$ MHV amplitude we define  $\mathcal{N}$  to be the set of all of the particles, i.e.  $\mathcal{N} := \{1, \dots, n\}$ . We will refer to the following equations as the ‘general  $d$  scattering equations’ [13]

$$\sum_{\substack{j \in \mathcal{N} \\ j \neq i}} \frac{k_i \cdot k_j}{s_i - s_j} = 0, \quad (2.1)$$

where  $k_i$  are a set of  $n$ -point null momenta obeying momentum conservation, and  $s_i$  are points on the Riemann sphere. In the general  $d$  formalism, tree-level amplitudes can then be

calculated as integrals of some integrand  $f(s)$  over delta functions enforcing these equations,

$$\mathcal{A}_n = \int \frac{d^n s}{\text{SL}(2)} \prod_{i \in \mathcal{N}} ' \delta \left( \sum_{\substack{j \in \mathcal{N} \\ j \neq i}} \frac{k_i \cdot k_j}{s_i - s_j} \right) f(s). \quad (2.2)$$

This type of integral arises from the calculation of a correlation function on a string worldsheet, and hence we will refer to the  $s$  as worldsheet variables. The equations have the standard  $\text{SL}(2)$  symmetry arising from global conformal transformations on the worldsheet, which reduces the number of integration variables by three. To balance this, three delta functions are removed and a corresponding Jacobian factor added, as denoted by the notation  $\prod_{i \in \mathcal{N}} '$ . Any valid integrand  $f$  must be covariant under this  $\text{SL}(2)$  symmetry, along with any other symmetries inherited from the corresponding amplitude. The number of integrations matches the number of delta functions, and hence this integral simply instructs us to sum the integrand times the relevant Jacobian factor over the  $(n-3)!$  solutions to the scattering equations. In this way, calculating tree-level scattering for massless particles in many theories has been reduced to solving a set of algebraic equations. Further details can be found in [12, 13].

In four dimensions we have additional structure due to the factorization of the Lorentz group  $\text{SO}(3,1) = \text{SL}(2)_L \times \text{SL}(2)_R$ , and the corresponding MHV classification which splits amplitudes up into different sectors depending on the number of negative helicity particles scattered. These additional structures can be described in terms of the spinor-helicity formalism, reviewed in [41, 42]. To clarify the notation, we define angle and square brackets such that  $|i\rangle \in \text{SL}(2)_L$  and  $|i] \in \text{SL}(2)_R$  describing the external momenta. Then  $k_i = |i\rangle \langle i| := |i\rangle \otimes \langle i|$  is the explicitly null external momentum on leg  $i$ . We form antisymmetric  $\text{SL}(2)$  invariant brackets as  $\langle ij \rangle := \det(|i\rangle |j\rangle)$  and  $[ij] := \det(|i] |j])$ .

We work with supersymmetry in an on-shell superspace via the Grassmann variables  $\eta_i^A$ , where  $A$  is an R-symmetry index and  $\eta_i^A$  are related to the supermomenta by  $q_i^A = |i\rangle \eta_i^A$ . Amplitudes in supersymmetric theories are most concisely described as a supersymmetry covariant superamplitude. This corresponds to scattering on-shell supermultiplets of the theory, and the superamplitude is a Grassmann expansion in terms of the  $\eta_i^A$ . The superamplitude will be a Grassmann function with some well-specified weight in  $\eta$  variables  $N_G$ . Amplitudes where  $N_G$  is not a multiple of  $\mathcal{N}$  are zero, and from this we define the Grassmann degree  $k_G := \frac{N_G}{\mathcal{N}}$  of the amplitude. Component amplitudes for individual states can be extracted by integrating against  $N_G$  relevant  $\eta$  variables. A review of these techniques can be found in [41, 42], and the extension to non-maximal supersymmetry is discussed in detail in [43].

As an example consider  $\mathcal{N} = 4$  super Yang-Mills with supermultiplet  $\Phi^\pm = g^- \eta_1 \eta_2 \eta_3 \eta_4 + \eta_I \eta_J \eta_K \psi^{-IJK} + \eta_I \eta_J \phi^{IJ} + \eta_I \psi^{+I} + g^+$ . The MHV superamplitude  $\mathcal{A}(++--)$  has  $k_G = 2$ , and for maximal supersymmetry the positive and negative supermultiplet are the same and hence we can choose which legs are  $+$  and  $-$ . For non-maximal SUSY the choice of assignment of superfields to each leg results in different superamplitudes. To extract a mixed fermion-gluon four point component amplitude, we read off the relevant  $\eta$

from the supermultiplet, and integrate the superamplitude against these Grassmann variables;

$$\mathcal{A}(g^-\psi^{-IJK}g^+\psi^{+L}) = \int d\eta_1^1 d\eta_1^2 d\eta_1^3 d\eta_1^4 d\eta_2^I d\eta_2^J d\eta_2^K d\eta_4^L \mathcal{A}(++--). \quad (2.3)$$

The 4D scattering equations are a refinement of the general  $d$  scattering equations which depend on the MHV degree as well as the number of particles [21], and calculate supersymmetry covariant expressions. They make use of spinor-helicity notation and are written in terms of the angle and square bracket spinors rather than momentum dot products. This introduces an extra variable  $t_i$  into the system for each particle which is related to the little group scaling of the spinor variables. We will think of the  $t_i$  as additional worldsheet variables, promoting each worldsheet coordinate from a point on the Riemann sphere to a point in  $\mathbb{C}^2$ . We will use different coordinates for the worldsheet variables as  $\sigma_i = \frac{1}{t_i} \begin{pmatrix} 1 \\ s_i \end{pmatrix} = \begin{pmatrix} \sigma_i^1 \\ \sigma_i^2 \end{pmatrix} \in \mathbb{C}^2$ . Combining all of worldsheet variables together as a matrix  $\sigma \in \mathbb{C}^{2 \times n}$ , we will work with minors of this matrix defined as  $(ij) := \det(\sigma_i \sigma_j) = \frac{s_i - s_j}{t_i t_j} = \sigma_i^1 \sigma_j^2 - \sigma_i^2 \sigma_j^1$ .

The external particles are now grouped into two different sets  $L$  and  $R$  with  $L \sqcup R = \mathcal{N}$  to respect the MHV degree of the amplitude. For Yang-Mills theory and Einstein gravity, the left set will correspond to the set of negative helicity particles (or more generally, those particles that sit in the negative helicity super-multiplet of the theory), and the right set will correspond to the positive helicity particles (resp. positive supermultiplet). In these cases the MHV degree  $k$  of the amplitude will be the size of the left set,  $k := |L|$ . More generally we can define an integrand in terms of some abstract left set, and indeed for example conformal supergravity amplitudes can have negative helicity particles in the left or the right sets, which we explain shortly.

We will refer to the following equations as the ‘4D scattering equations’ [21]. They are specified by an integer  $n$  and a left set  $L$ .

$$\begin{aligned} \tilde{E}_l &:= |l] - \sum_{r \in R} \frac{|r]}{(lr)} = 0, \quad l \in L & E_r &:= |r\rangle - \sum_{l \in L} \frac{|l\rangle}{(rl)} = 0 \quad r \in R \\ \delta^{2 \times n}(SE_L^n) &:= \prod_{l \in L} \delta^2(\tilde{E}_l) \prod_{r \in R} \delta^2(E_r). \end{aligned} \quad (2.4)$$

As in the general  $d$  case, we can then write amplitudes for many different theories as integrals of some integrand  $f$  over these scattering equation delta functions;

$$\mathcal{A}_{n,L} = \int \frac{d^{2 \times n} \sigma}{\text{GL}(2)} \delta^{2 \times n}(SE_L^n) f(\sigma, |i\rangle, |i]).$$

We will also analyse the integrands for some specific theories. Integrands are currently known for Einstein gravity and Yang-Mills theory with any valid number of supersymmetries [21], and for graviton multiplets in  $\mathcal{N} = 4$  conformal supergravity [23]. The supersymmetry structure is encoded by a set of fermionic delta functions of the form  $\delta^{\mathcal{N}}\left(\eta_l - \sum_{r \in R} \frac{\eta_r}{(lr)}\right)$ , which can be thought of as scattering equations for the Grassmann on-shell superspace variables. In this paper we include the fermionic delta functions in

the integrand, rather than as part of the scattering equations. The integrands for these theories are

$$\begin{aligned}
 f_{\text{sYM}}(\sigma, |i\rangle, |i], \eta_i) &:= \prod_{l \in L} \delta^{\mathcal{N}} \left( \eta_l - \sum_{r \in R} \frac{\eta_r}{(lr)} \right) \frac{1}{\prod_{i \in N} (i \ i+1)} \\
 f_{\text{SUGRA}}(\sigma, |i\rangle, |i], \eta_i) &:= \prod_{l \in L} \delta^{\mathcal{N}} \left( \eta_l - \sum_{r \in R} \frac{\eta_r}{(lr)} \right) \det' \mathcal{H} \det' \tilde{\mathcal{H}} \\
 f_{\text{CSG}}(\sigma, |i\rangle, |i], \eta_i) &:= \prod_{l \in L} \delta^4 \left( \eta_l - \sum_{r \in R} \frac{\eta_r}{(lr)} \right) \prod_{l^- \in L \cap \Phi^-} \mathcal{H}_{l^-} \prod_{l^+ \in L \cap \Phi^+} \tilde{\mathcal{F}}_{l^+} \prod_{r^- \in R \cap \Phi^-} \mathcal{F}_{r^-} \prod_{r^+ \in R \cap \Phi^+} \tilde{\mathcal{H}}_{r^+},
 \end{aligned} \tag{2.5}$$

where  $\det'$  is an instruction to remove one row and column from the matrices before taking the determinant.  $\mathcal{H}$  is the Hodges matrix and  $\tilde{\mathcal{H}}$  the dual Hodges matrix, defined as

$$\begin{aligned}
 \mathcal{H}_{ll} &:= - \sum_{l' \neq l \in L} \mathcal{H}_{ll'}, & l \in L & & \mathcal{H}_{ll'} &:= \frac{\langle ll' \rangle}{(ll')}, & l \neq l' \in L \\
 \tilde{\mathcal{H}}_{rr} &:= - \sum_{r' \neq r} \tilde{\mathcal{H}}_{rr'}, & r \in R & & \tilde{\mathcal{H}}_{rr'} &:= \frac{[rr']}{(rr')}, & r \neq r' \in R.
 \end{aligned} \tag{2.6}$$

In conformal supergravity the left set  $L$  does not correspond to the set of negative helicity superfields, which we denote  $\Phi^-$  (with  $\Phi^+$  the positive helicity superfields;  $\Phi^- \sqcup \Phi^+ = \mathcal{N}$ ). The left set  $L$  of the scattering equations has  $|L| = k_G$ , where  $k_G$  is the Grassmann degree of the superamplitude considered, and the choice of  $L$  does not affect the amplitude. The factors  $\mathcal{H}$ ,  $\mathcal{F}$ ,  $\tilde{\mathcal{H}}$  and  $\tilde{\mathcal{F}}$  in conformal supergravity are generalisations of the gravitational inverse soft factor outside of the MHV sector. The  $\mathcal{H}$  factors are the diagonal elements of the Hodges matrices, and the  $\mathcal{F}$  are given by

$$\tilde{\mathcal{F}}_l := \sum_{r < r' \in R} \frac{[rr'] (rr')}{(lr)^2 (lr')^2} \qquad \mathcal{F}_r := \sum_{l < l' \in L} \frac{\langle ll' \rangle (ll')}{(rl)^2 (rl')^2}. \tag{2.7}$$

As shown in [28], there are  $A(n, k) = \langle \frac{n-3}{k-2} \rangle$  solutions to the  $n$ -point  $N^{k-2}$ MHV scattering equations.  $\langle \frac{n}{k} \rangle$  are the Eulerian numbers [44], tabulated in table 1. The inductive proof of the number of solutions finds that  $A(n, k)$  can be broken down by taking either a left set particle soft to reduce to an amplitude of the form  $A(n-1, k-1)$  or a right set particle soft to reduce to  $A(n-1, k)$ . Each of these solutions then has a given multiplicity such that

$$A(n, k) = (n - k - 1)A(n - 1, k - 1) + (k - 1)A(n - 1, k), \tag{2.8}$$

with one solution in the MHV and  $\overline{\text{MHV}}$  sectors for each  $n$ , so that  $A(n, 2) = A(n, n-2) = 1$ .

### 3 Calculating amplitudes with the 4D scattering equations

In this section we describe the analytical results necessary for solving the scattering equations to calculate amplitudes given a certain integrand. These techniques will then be used



explicitly to find amplitudes numerically and analytically in the package `treeamps4dJAF`. Detailed proofs are given in appendix B.

As shown in equation (2.4), at  $n$  points there are  $2n$  4D scattering equations depending on  $2n$  worldsheet  $\sigma$  variables. There is a  $\text{GL}(2)$  symmetry on the worldsheet which acts as inhomogeneously on the left and right set of worldsheet coordinates for  $G \in \text{GL}(2)$  as

$$\sigma_l \rightarrow G \sigma_l, \quad l \in L \quad \sigma_r \rightarrow \frac{G}{\det G} \sigma_r, \quad r \in R. \quad (3.1)$$

Under this  $\text{GL}(2)$  any minor of the form  $(lr)$  remains invariant, and hence the scattering equations are invariant. Fixing this gauge symmetry leaves  $2n - 4$  remaining degrees of freedom. Generally in this work we will restrict to gauge transformations specified by two particle labels  $i, j$  which fix  $\sigma_i = (\frac{1}{0})$  and  $\sigma_j = (\frac{0}{1})$ , and refer to this operation as ‘gauge-fixing particles  $i$  and  $j$ ’. As the  $\text{GL}(2)$  transformations act inhomogeneously on the left and right set, we can only fix either two left or two right particles this way. The details of gauge-fixing and the symmetries of the equations are explained in appendix B.2.

The system now appears to be over specified, and we must remove four equations. Two spinor equations  $i$  and  $j$  either from the left set or from the right set can be reduced to a momentum conserving delta function on support of the other scattering equations, as proved in appendix B.3. We then refer to having ‘deleted particles  $i$  and  $j$ ’, and the remaining equations are a well-specified set of  $2n - 4$  equations in  $2n - 4$  variables.

The number of integrations in equation (2) is the same as the number of delta functions, and hence the integrations instruct us to sum over all of the solutions of the scattering equations. Deleting equations  $l, l' \in L$  we can calculate the Jacobian of the remaining equations to solve the delta function integrals as follows

$$\mathcal{A}_{n,L} = \int \frac{d^{2 \times n} \sigma}{\text{GL}(2)} \delta^{2 \times n} (SE_L^n) f(\sigma) = \delta^4(P) \sum_{\sigma_{\text{sol}} \in \text{solutions}} \frac{f(\sigma_{\text{sol}})}{\langle ll' \rangle^{-2} \det(J_L^{n, ll'}(\sigma_{\text{sol}}))}. \quad (3.2)$$

where  $J_L^n$  is the Jacobian of the scattering equations with respect to the sigma variables, and the superscript  $l, l'$  refers to removing four rows and columns corresponding to the two particle labels from the matrix. Details of the Jacobian to the scattering equations are explained in appendix B.5. It is now a well-formulated problem to solve the scattering equations and sum a theory dependent integrand  $f$  over the full set of solutions to produce an amplitude.

Calculating MHV amplitudes is generally more simple than calculating  $N^{k-2}\text{MHV}$  amplitudes, and we find this simplicity to be reflected in the structure of the 4D scattering equations. Analytical solutions to the 4D scattering equations are not known for general MHV degree, but in the MHV sector we can construct analytical solutions. First consider the case where the left set is  $L = \{1, 2\}$ . Then the MHV equations become

$$\delta^{2 \times n} (SE_{\{1,2\}}^n) = \delta^2 \left( [1] - \sum_{r \in R} \frac{[r]}{(1r)} \right) \delta^2 \left( [2] - \sum_{r \in R} \frac{[r]}{(2r)} \right) \prod_{r \in R} \delta^2 \left( |r\rangle - \frac{|1\rangle}{(1r)} + \frac{|2\rangle}{(2r)} \right) \quad (3.3)$$

The most obvious choice of equations to remove in this case is the two left set equations which become the overall momentum conservation delta function as detailed in appendix B.3, and we recover a Jacobian of  $\langle 12 \rangle^2$ . Similarly we gauge-fix particles 1 and 2 to the identity in the Grassmannian, and arrive at the following form for each delta function of the right-set equations, which are solved by a Schouten identity;

$$\delta^2 \left( |r\rangle - \frac{|1\rangle}{\sigma_r^2} - \frac{|2\rangle}{\sigma_r^1} \right) = \frac{\langle 12 \rangle^3}{\langle 1r \rangle^2 \langle 2r \rangle^2} \delta \left( \sigma_r^1 - \frac{\langle 12 \rangle}{\langle r1 \rangle} \right) \delta \left( \sigma_r^2 - \frac{\langle 12 \rangle}{\langle r2 \rangle} \right), \quad r \in R. \quad (3.4)$$

The full MHV solution along with its minors and the associated expression for the Jacobian of the delta functions is

$$\begin{aligned} \sigma_{\text{MHV}} &= \begin{pmatrix} 1 & 0 & \frac{\langle 12 \rangle}{\langle 31 \rangle} & \cdots & \frac{\langle 12 \rangle}{\langle n1 \rangle} \\ 0 & 1 & \frac{\langle 12 \rangle}{\langle 32 \rangle} & \cdots & \frac{\langle 12 \rangle}{\langle n2 \rangle} \end{pmatrix} \\ \delta^{2 \times n} \left( SE_{\{1,2\}}^n \right) &= \langle 12 \rangle^2 \prod_{r \in R} \frac{\langle 12 \rangle^3}{\langle 1r \rangle^2 \langle 2r \rangle^2} \delta^4(P) \delta^{2n-4}(\sigma - \sigma_{\text{MHV}}) \\ (12)_{\text{MHV}} &= 1 \quad (rr')_{\text{MHV}} = \frac{\langle 12 \rangle^3 \langle rr' \rangle}{\langle 1r \rangle \langle 1r' \rangle \langle 2r \rangle \langle 2r' \rangle} \\ (1r)_{\text{MHV}} &= \frac{\langle 12 \rangle}{\langle r2 \rangle} \quad (2r)_{\text{MHV}} = \frac{\langle 12 \rangle}{\langle 1r \rangle}. \end{aligned} \quad (3.5)$$

The Jacobian as calculated this way is in agreement with the calculation from appendix B.5, where we also consider the Jacobian for higher MHV degree.

Finding analytical solutions for generic kinematics outside of the MHV sector is currently an unsolved problem, apart from at 6 points NMHV where the scattering equations have four solutions. These solutions are found for the general  $d$  equations for  $d = 4$  in [45]. Abel's theorem states that there is no algebraic solution in terms of  $n$ th roots to a general polynomial equation of degree five or higher with arbitrary coefficients [46]. To find analytical  $N^{k-2}$ MHV solutions above 6 points NMHV some underlying structure would have to exist within the coefficients of the equations, otherwise general analytical solutions are excluded by Abel's theorem. Full sets of solutions can be found analytically for some specific choices of momenta [47].

Due to these difficulties in finding analytical solutions, calculating amplitudes by solving the scattering equations outside of the MHV sector is primarily a numerical problem, which we address in section 4.

One key strength of the scattering equation formalism is that once a full set of solutions are known, amplitudes can be calculated in any theory at relatively small computational cost. The relevant integrand is chosen, and the only necessary operation is to sum over the solutions. Solutions to the scattering equations are graded only by MHV degree and not by a specific choice of left set. This implies that there must exist some transformation on the worldsheet which can map an integrand supported on scattering equations for one left set into an integrand for a different left set of the same length. For example such a mapping will allow calculation of all 6 point NMHV gluon amplitudes in Yang Mills theory with

only one solution to the scattering equations; eg.  $\mathcal{A}(- - + - + -)$  with left set  $L = \{2, 4, 6\}$  and  $\mathcal{A}(- - - + + +)$  with left set  $L = \{1, 2, 3\}$ .

The following is an explicit co-ordinate transformation on the worldsheet which swaps two particles between the left and right sets of the scattering equations.<sup>1</sup> We single out two legs  $l_0 \in L$  and  $r_0 \in R$  which will be swapped.

$$\begin{aligned} \sigma_{l_0} &\rightarrow \sigma'_{l_0} = \sigma_{l_0} \frac{1}{(l_0 r_0)}, & \sigma_{r_0} &\rightarrow \sigma'_{r_0} = \sigma_{r_0} \frac{1}{(r_0 l_0)}, \\ \sigma_l &\rightarrow \sigma'_l = \sigma_l \frac{(ll_0)}{(lr_0)} \quad l \neq l_0 \in L, & \sigma_r &\rightarrow \sigma'_r = \sigma_r \frac{(rr_0)}{(rl_0)} \quad r \neq r_0 \in R. \end{aligned} \quad (3.6)$$

Under this transformation we find that the scattering equation integrand transforms as

$$\begin{aligned} \int \frac{d^{2 \times n} \sigma}{\text{GL}(2)} \delta^{2 \times n}(SE_L^n) f(\sigma) &\rightarrow \\ \int \frac{d^{2 \times n} \sigma}{\text{GL}(2)} \delta^{2 \times n}(SE_{L'}^n) f'(\sigma) &= \int \frac{d^{2 \times n} \sigma}{\text{GL}(2)} \delta^{2 \times n}(SE_{L'}^n) \prod_{l \neq l_0 \in L} \frac{(ll_0)^2}{(lr_0)^2} \prod_{r \neq r_0 \in R} \frac{(rr_0)^2}{(rl_0)^2} \frac{f(\sigma')}{(l_0 r_0)^8} \end{aligned} \quad (3.7)$$

where  $L'$  has  $l_0$  swapped with  $r_0$ . We now have an explicit transformation for how to calculate a new integrand  $f'$  for a swap of the choice of left sets for the scattering equations. Details of the transformation under this mapping are given in appendix B.4. Repeatedly applying this transformation can be used to reassign any left set.

## 4 Numerical methods

The 4D scattering equations can be thought of as  $2n - 4$  equations with variables in the Grassmannian  $\text{Gr}(2, n)$ , as explained in appendix B.2. The equations are parametrised by a set of spinors obeying momentum conservation. A solution to the scattering equations at  $n$  points  $N^{k-2}\text{MHV}$  will be a mapping from external data to  $\langle \frac{n-3}{k-2} \rangle$  points in  $\text{Gr}(2, n)$ . In the MHV case we give this mapping analytically in section 3, but finding analytical solutions for  $k > 2$  is complicated due to the combinatorially increasing number of solutions. A well-specified problem is to provide explicit numerical momenta, which will usually be randomly sampled, and to then solve the resulting equations numerically. CHY provide an inverse-soft type algorithm for finding individual numerical solutions to the general  $d$  equations [12], but there are difficulties in constructing the full set of solutions in this way which we discuss in section 4.1. We provide an explicit algorithm which given a set of numerical momenta samples random numerical points in  $\text{Gr}(2, n)$  to find solutions to the equations stochastically. Algorithms of this type are known as Monte Carlo algorithms. Monte Carlo methods in high energy physics are well studied [48, 49] and their application to solving non-linear algebraic equations is straightforward.

With enough computing power and time, any non-linear system of equations can be solved by Monte Carlo algorithm. The two key questions to address are when to stop the

---

<sup>1</sup>We thank Paul Heslop for suggesting this transformation.

$n$	$\langle \frac{n-3}{k-2} \rangle$
4	1
5	1 1
6	1 4 1
7	1 11 11 1
8	1 26 66 26 1
9	1 57 303 302 57 1
10	1 120 1191 2416 1191 120 1

**Table 1.** Eulerian numbers of solutions to the 4D scattering equations.

algorithm, and what distribution to sample the initial guess points from. The scattering equations are well-suited for solution in this way because the number of solutions is known, which gives a clear stopping condition. We address the sampling question in section 4.2. Finding a set of solutions this way is stochastic and can take a long time, with time complexity now distributed as a random variable which depends on  $n$  and  $k$ . The expectation of the time complexity increases as  $n$  increases and as  $k$  moves towards  $\lfloor \frac{n}{2} \rfloor$ . One advantage of the 4D formalism that makes it better suited for solution by Monte Carlo algorithm is that the  $(n-3)!$  solutions are broken down into Eulerian numbers of solutions, tabulated in table 1. This means that the algorithm can stop after finding a smaller number of solutions than in general dimensions.

Once a full set of numerical solutions along with the corresponding momenta and left set are known for a given number of points and MHV degree, they can then be used to calculate amplitudes in different theories for a selection of different external states by substituting different integrands into the sum over solutions. Solutions up to 12 points NMHV and 9 points in all MHV sectors are currently accessible to the algorithms of `treeamps4dJAF`, and we tabulate full solution sets with rational external data for these cases in the accompanying data file `SolutionLookupTable.csv`.

Tree-level amplitudes are all rational functions of external momenta, and hence for rational numerical external data they will be a rational number. The solutions to the scattering equations are in general not rational numbers, but given a set of rational kinematics we can calculate to very high precision at relatively low computational cost via deterministic algorithm once all solutions are known. It is then possible rationalize to the closest rational number to give exact numerical results for the amplitude. We provide support for this type of calculation in `treeamps4dJAF`.

#### 4.1 Difficulties with CHY's inverse soft algorithm

One proposed algorithm to find numerical solutions to the general  $d$  equations is that of CHY, which takes one of the momenta soft with parameter  $\epsilon$  to reduce the equations from  $n$  points down to  $n-1$  points [12]. The soft parameter is then reintroduced, and the soft equation at  $O(\epsilon)$  is solved for each of the  $(n-4)!$  solutions to the  $n-1$  point equations. The solutions then have a multiplicity of  $n-3$ , and these points are input back into the system with  $\epsilon$  moving slowly up from 0 to 1. As this algorithm involves slowly bringing

the soft parameter back to the full  $n$  point system, it is referred to as an inverse soft algorithm.  $(n-3)!$  solutions to the  $n$  point equations will be found in this way, but there is no guarantee that all of these solutions will be distinct, and hence they will not necessarily cover the full solution space.

The inverse soft algorithm is based on an inductive argument for counting the number total number of solutions, and we can extend it to the 4D case using the analogous 4D solution counting argument, which is reviewed in section 2. We take one soft parameter  $\epsilon$  for a left set particle  $1 \in L$  so that  $|1\rangle \rightarrow \epsilon|1\rangle$ , and a further parameter  $\tilde{\epsilon}$  for a right set particle  $n \in R$  so that  $|n\rangle \rightarrow \tilde{\epsilon}|n\rangle$ . The 4D scattering equations become

$$\begin{aligned} |1\rangle - \sum_{r \neq n \in R} \frac{|r\rangle}{(1r)} - \tilde{\epsilon} \frac{|n\rangle}{(1n)} &= 0 & |n\rangle - \epsilon \frac{|1\rangle}{(n1)} - \sum_{l \neq 1 \in L} \frac{|l\rangle}{(nl)} &= 0 \\ |l\rangle - \sum_{r \neq n \in R} \frac{|r\rangle}{(lr)} - \tilde{\epsilon} \frac{|n\rangle}{(ln)} &= 0, \quad l \neq 1 \in L & |r\rangle - \epsilon \frac{|1\rangle}{(r1)} - \sum_{l \neq 1 \in L} \frac{|l\rangle}{(rl)} &= 0 \quad r \neq n \in R. \end{aligned} \quad (4.1)$$

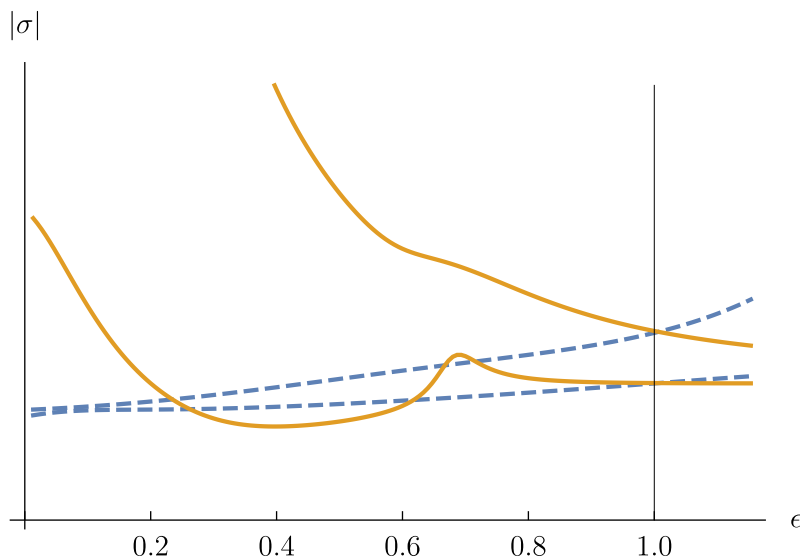
We can see that worldsheet variable for the particle in the soft limit decouples, and the equations reduce to  $n-1$  point  $N^{k-2}$ MHV equations when  $\epsilon = 1, \tilde{\epsilon} \rightarrow 0$  and  $n-1$  point  $N^{k-3}$ MHV equations when  $\tilde{\epsilon} = 1, \epsilon \rightarrow 0$ . Evaluated on the solution to the lower point equations, the remaining equation for the particle that decoupled gives the multiplicity for each solution, as shown in equation (2.8).

As detailed in [12] this method is sufficient to produce individual solutions for a specific  $n$  and  $k$ , but we find difficulties when trying to construct all of the solutions in this way. In four dimensions we find that two of the different solutions constructed from a lower point amplitude can converge to the same higher point solution, as shown at 6 points NMHV in figure 1. Hence the maximum number of solutions this algorithm can find is  $\binom{n-3}{k-2}$ , and generically it does not find all of the solutions.

MHV and  $\overline{\text{MHV}}$  solutions are known analytically, and the first non MHV case is at 6 points NMHV, with four solutions. The  $\epsilon$  soft limit gives a 5 point  $\overline{\text{MHV}}$  amplitude, and the  $\tilde{\epsilon}$  soft limit produces a 5 point MHV amplitude. The soft limit equations both have two solutions. Figure 1 describes the norm of the MHV solutions as they evolve from  $\epsilon = 0$  up to  $\epsilon = 1$  in blue, and the  $\overline{\text{MHV}}$  solutions from  $\tilde{\epsilon} = 0$  up to  $\tilde{\epsilon} = 1$  in yellow.

We define a matrix norm on the solutions by taking the standard norm on  $\mathbb{C}^n$ , after flattening the  $2 \times 4$  worldsheet matrix with gauge-fixed rows deleted down to  $\mathbb{C}^8$ . Lines crossing on the plot in this description is not sufficient for the solutions to be equal, and indeed where the lines cross for  $\epsilon \in [0, 1]$  the solutions are not equal. At  $\epsilon = 1$  the solutions converge. Interestingly when we continue the algorithm to run for  $\epsilon$  slightly larger than the 1 we see that the solutions separate again.

In the case shown in figure 1, both of the solutions from 5 points MHV collide separately with two individual solutions from 5 points  $\overline{\text{MHV}}$ . In general for different randomly selected numerical momenta at 6 points NMHV we find zero, one or two pairs of solutions converging. The number of pairs that converge appears to be random based on the selection of random momenta. Solution crashing is a ubiquitous phenomenon; it occurs for nearly all choices and the difficulty is rather to find cases where the solutions do not crash than to find cases where they do.



**Figure 1.** Convergent solutions at 6 points NMHV using CHY's inverse soft algorithm. Orange solid lines are solutions coming from 5 points  $\overline{\text{MHV}}$  and blue dashed lines are from 5 points MHV.

These difficulties with the inverse soft algorithm inspire developing new methods, and we solve this problem using a Monte Carlo algorithm.

## 4.2 Monte Carlo algorithm

The Monte Carlo equation solving algorithm in `treeamps4dJAF` is implemented via `NSolveMonteCarlo`. Many initial random points are sampled from a distribution described below, and chosen as the initial conditions for a `FindRoot` calculation. These initial calls to `FindRoot` run for many iterations, and stop after only one digit of precision is met. Most initial guess points will be far from a solution, and will not converge to 1 digit of precision by the specified number of iterations. Those which do not converge are discarded, and the ones that do converge go back into `FindRoot` up to a higher precision. These points are now solutions, which are compared with a list of all currently found solutions and duplicates are discarded. The algorithm stops when a suitable stopping condition is met, which may be after a specified amount of time or number of iterations, or when enough solutions are found. A pseudo-code for this algorithm is

**function** `NSolveMonteCarlo`(*equations*, *variables*)

Compile equations and Jacobian down to C code for faster evaluation

**while** a stopping condition is not met **do**

Sample 100 initial solution points from a specified distribution

Run `FindRoot` on each point. Stop after 1000 iterations, or when a point solves the equations to 1 digit precision

Run `FindRoot` to higher precision for points that solved the equations

Compare solutions to a specified precision and discard any duplicate solutions found

$n$	$k$	$\langle \frac{n-3}{k-2} \rangle$	$N_{\text{runs}}$	$N_{\sigma}$	$\gamma_{\sigma}$	$\text{MAD}_{\sigma}$	$\bar{t}$	$\text{STD}_t$
6	3	4	7559	483776	0.448	0.462	1.82 s	1.23 s
7	3	11	1047	230340	0.433	0.448	12.5 s	12.2 s
8	3	26	1001	624624	0.443	0.461	1. min	50.1 s
9	3	57	57	90972	0.408	0.427	22.9 min	15.8 min
10	3	120	9	34560	0.423	0.427	4.33 hr	3.65 hr
11	3	247	2	17784	0.508	0.508	10.7 hr	1.57 hr
7	4	11	1001	220220	0.482	0.485	39.8 s	1.49 min
8	4	66	53	83952	0.439	0.447	31. min	29.9 min
9	4	302	2	16912	0.397	0.403	28.7 hr	2.67 hr
8	5	26	328	204672	0.539	0.533	3.45 min	3.02 min
9	6	57	51	81396	0.665	0.642	21.2 min	17.6 min
10	7	120	17	65248	0.698	0.702	59.8 min	34.9 min
11	8	247	2	17784	0.627	0.61	11.5 hr	7.81 hr
12	9	502	1	20080	1.71	1.76	22.3 hr	—

**Table 2.** Statistical summary of distribution of solutions to 4D scattering equations and timings of `NSolveMonteCarlo` algorithm.  $N_{\text{runs}}$  is the number of different set of numerical momenta used,  $N_{\sigma} = (4n - 8) \langle \frac{n-3}{k-2} \rangle N_{\text{runs}}$  is the number of solution points,  $\gamma_{\sigma}$  is the scaling parameter of the fitted Cauchy distribution and  $\text{MAD}_{\sigma}$  is the median absolute deviation.  $\bar{t}$  is the average time and  $\text{STD}_t$  is the standard deviation. Note that when  $N_{\text{runs}}$  is small solution point statistics may not be reliable even though  $N_{\sigma}$  is large, as they come from a small number of different choices of numerical momenta.

**end while**

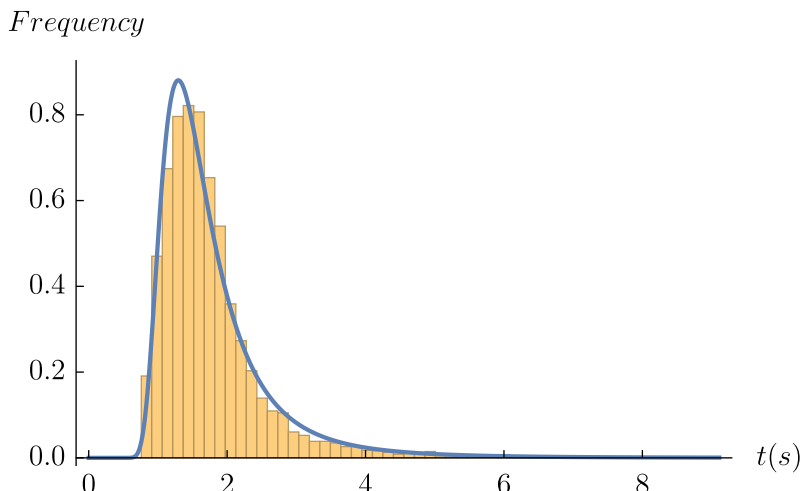
Return the solutions

**end function**

`NSolveMonteCarlo` is tailored to the 4D scattering equations in the function `NSolveScatteringEquations4D`. This specifies the stopping condition to be when all of the Eulerian numbers of solutions are found, and selects an appropriate distribution to sample the random points from. Table 2 gives a statistical analysis of the time complexity of the algorithm<sup>2</sup> and of the distribution of solution points for all currently accessible  $n$  and  $k$ . Figure 2 expands on this analysis for 6 points NMHV, giving a histogram of the timings. Timings are positively skewed with a similar shape for other  $n$  and  $k$ . Based on this, the algorithm can currently handle at most around 500 total solutions. Accessing the next cases would require around 1000 solutions, at 10 points  $\text{N}^2\text{MHV}$  and 13 points NMHV.

We now analyse the sampling of the initial points, and explain which distribution to sample from for most efficient results. We solve momentum conservation in terms of the spinors [1] and [2], and delete equations 1 and 2 so that these spinors do not appear in the equations to be solved. This way every random numerical spinor generated is uncon-

<sup>2</sup>All timings were calculated on a Linux desktop computer with 3.30GHz Intel(R) Core(TM) i7-5820K processor, and vary depending on what other processes were running during evaluation of the algorithm.



**Figure 2.** Histogram of timings for finding solutions for 7559 sets of solutions to the 4D scattering equations at 6 pts NMHV using `NSolveScatteringEquations4D`. The blue curve is `FrechetDistribution[3.7,1.6,-0.2]`, as a best fit by MATHEMATICA.

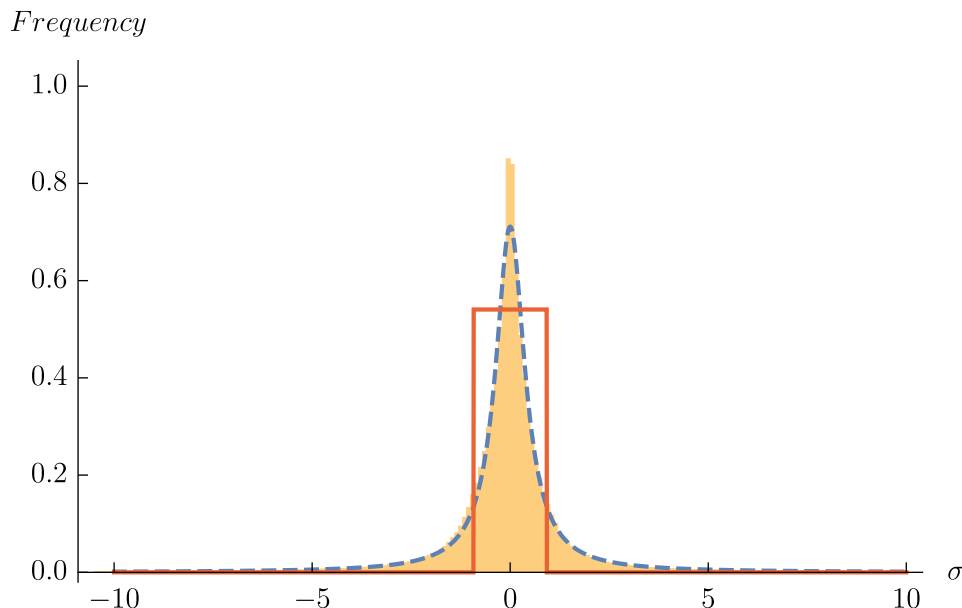
strained, and hence comes from the same uniform distribution. Note that the statistics of the solution points is gauge dependent, and as we always gauge fix particles 1 and 2 it is not possible to make a direct comparison between data points with  $k = k'$  and with  $k = n - k'$ , even though these cases at first glance should be symmetrical.

We perform a statistical analysis at 6 points NMHV, as guided by our analytical understanding in the MHV sector. A solution at 6 points NMHV can be considered as a matrix in  $\mathbb{C}^{2 \times 4}$  after removing gauge-fixed columns. We project this down to a vector of real numbers in  $\mathbb{R}^{16}$ , and hence collect 48 real numbers for each set of numerical momenta which we refer to as ‘solution points’. We then run the algorithm for statistically many sets of random numerical momenta, and collect all of the solution points together into one dataset. A histogram of this data is plotted in figure 3. The data are best fitted by a Cauchy distribution, for which the probability density function has the form

$$\mathbb{P}(x; x_0, \gamma) = \frac{1}{\pi\gamma} \left( \frac{\gamma^2}{(x - x_0)^2 + \gamma^2} \right). \quad (4.2)$$

The data are symmetrically distributed around 0, and hence we see that the location parameter  $x_0 = 0$ . This leaves only one remaining parameter  $\gamma$ , which is tabulated for some cases in table 2. Apriori we would expect  $\gamma$  be a function of  $n, k$  and the size of the uniform distribution from which the random momenta are sampled. We can use some intuition from the analytical result in the MHV sector to reduce this down to just  $\gamma = \gamma(n, k)$ . Solution points in the MHV sector have a form such as  $\text{Re}(\frac{\langle 12 \rangle}{\langle 1r \rangle})$ , as shown in section 3. Hence the size of the uniform distribution from which the random events are sampled will divide out statistically, and will not affect the distribution of the solution points. This independence is guaranteed for higher MHV degrees as solutions has mass dimension 0, and spinors have mass dimension  $\frac{1}{2}$ . Performing a statistical analysis of numerical solutions to the scattering equations outside of the MHV sector also verifies these properties, for example as shown





**Figure 3.** Histogram of 43776 solution points at 6 points NMHV taken from 7559 different random momenta. The blue dashed curve is `CauchyDistribution[0,0.447]`, and the red solid curve is an approximation with the uniform distribution  $\mathcal{U}([-0.894, 0.894])$ .

in table 2 where different values for  $\gamma$  are tabulated. Our data in table 2 show that  $\gamma$  is insensitive to  $k$  also. We find this independence of  $\gamma$  on  $n$  and  $k$  to be intriguing and counter-intuitive.

From this analysis, it seems clear that we should sample the initial points from the relevant Cauchy distribution. However, very many initial points must be sampled for each iteration of the algorithm, and sampling points from Cauchy distribution is significantly slower than sampling from a uniform distribution. We find in practice that it is more efficient to approximate this Cauchy distribution with a uniform distribution. To approximate a normal distribution with a uniform distribution, one could choose a symmetric range with a width which is a small multiple of the standard deviation, eg.  $\mathcal{U}([-2\sigma, 2\sigma])$ . This poses a problem with the Cauchy distribution as it has infinite standard deviation. We use instead the median absolute deviation (MAD), which is roughly equivalent to the  $\gamma$  parameter of the Cauchy distribution, and is tabulated for different  $n$  and  $k$  in table 2. To solve the equations for a given  $n$  and  $k$ , `NSolveScatteringEquations4D` uses the tabulated MAD values from table 2 to sample initial points from  $\mathcal{U}([-2\text{MAD}, 2\text{MAD}])$ .

When finding solutions by Monte Carlo algorithm, a large percentage ( $\sim \frac{3}{4}$ ) of the solutions tend to be found comparatively quickly, and it can take a long time to find the remaining solutions. It is possible that sampling a certain percentage of the solutions from the approximated uniform distribution and then sampling the remaining solutions from the relevant Cauchy distribution could help to overcome this problem. We leave this for future work.

### 4.3 Efficient component amplitude extraction

Extracting individual component amplitudes from a super amplitude written as an expansion in Grassmann parameters is a well specified and well understood operation, as explained in section 2. Explicitly evaluating these calculations on the computer is not as simple as the operation itself might suggest. The most naïve application is to expand out the Grassmann super amplitude in terms of each of its factors, and to then throw away any of the terms which are equal to zero either due to  $\eta^2 = 0$  or because do not match the integration measure. This algorithm is prohibitive in terms of memory usage in the computer. For example a product of  $m$  factors each with a sum of  $m$  terms of Grassmann numbers will result in a total of  $m^m$  terms when expanding out naively, nearly all of which are zero. This motivates finding a more efficient algorithm.

In the scattering equation formalism, the Grassmann delta functions are always written in the form  $\prod_{l \in L} \delta^{\mathcal{N}} \left( \eta_l - \sum_{r \in R} \frac{\eta_r}{(lr)} \right)$ . We can see that integrals of this expression over a subset of the  $\eta$  variables of dimension  $k\mathcal{N}$  is a special case of a Grassmann integral of the form  $I_G := \int d^m \eta \prod_{i=1}^m (\sum_{j=1}^m A_{ij} \eta_j)$ ; in our real use case the matrix  $A$  has some components given by worldsheet minors, and the rest zero. Grassmann integrals of this type can be evaluated in an especially neat analytical form, and we find that  $I_G = \det(A)$ . Numerical computation of determinants is generally implemented with an algorithm of time complexity around  $O(m^3)$ , and hence we can extract Grassmann components in a very efficient way using this formula. Grassmann integration of functions of this form then consists simply of assigning the correct values to the matrix  $A$  and calculating its determinant. Any product of Grassmann delta functions takes this form, and specifically the fermionic delta functions of the scattering equations, as well as many other standard representations for superamplitudes.

We can improve the RAM and time efficiency of this algorithm further for numerical computations by storing the matrix as a sparse array, where all elements are assumed to be zero unless they are specified (in our usage case as worldsheet minors). We can then hold the evaluation of the determinant until explicit numerical values are substituted in for the worldsheet minors, and then calculate the determinant of a numerical matrix. Calculation in this way is very efficient, saving the need to store large intermediate analytical expressions in the RAM.

## 5 Conclusions

In this paper we provide the necessary tools for solving the 4D scattering equations analytically in the MHV sector and numerically for higher MHV degree. We outline a Monte Carlo algorithm for solving the scattering equations numerically, and an algorithm for extracting component amplitudes from Grassmann delta functions efficiently. This allows computation of tree-level amplitudes in Yang-Mills theory and Einstein gravity with any number of super symmetries, and  $\mathcal{N} = 4$  conformal supergravity. Amplitudes in these theories can be calculated explicitly using the accompanying MATHEMATICA package `treeamps4dJAF`, which implements the algorithms described in this paper. When integrands are conjectured for new theories it will now be straightforward to test and evaluate them using the package.

We include some discussion of the key functions from the package in this paper, and in the supporting files we provide the MATHEMATICA package along with example code and full documentation, and a lookup table with solutions up to 12 points NMHV and 9 points in all MHV sectors.

A natural question to ask is whether these techniques are applicable to the general  $d$  scattering equations. It should be simple to map a full set of solutions at  $n$  points from the 4D scattering equations to the general  $d$  scattering equations for  $d = 4$ , which would allow calculating amplitudes for the wider variety of theories supported by these equations in four dimensions. Solving the general  $d$  equations directly using this algorithm is more difficult as they are not refined by helicity degree and all  $(n - 3)!$  solutions must be found. MATHEMATICA's inbuilt algorithms can already solve the general  $d$  equations numerically up to 9 points with a deterministic algorithm, so it would be necessary to improve on the efficiency of the Monte Carlo algorithm if it is to become a viable method for this problem.

`treeamps4dJAF` provides only a basic implementation of the Monte Carlo equation solving algorithm, and there are a number of different ways that it could be improved. `NSolveMonteCarlo` does not support parallel computation of different calls to `FindRoot`, and does not use an optimal algorithm for deciding whether solutions are duplicate. The sampling method used where all initial points are taken from an approximated uniform distribution could be improved on. An updated algorithm with these changes in MATHEMATICA should have significantly better behaved time complexity. Re-writing the core solution finding algorithm in a different programming language better suited to low-level numerical computation such as C++ could also increase efficiency dramatically. These changes should allow solution finding in the 4D case for higher  $n$  and  $k$ , and it is possible that they could allow solving the general  $d$  equations by Monte Carlo algorithm to become viable.

It is intriguing to find that the Cauchy distribution arises in the statistical analysis of the solutions to the 4D scattering equations, and especially to find that the parameter of the distribution is insensitive to changing  $n$  and  $k$ . It would be interesting to explore how the functional form of this distribution is related to the physics of the equations; the statistical analysis of this paper is dependent on choice of  $GL(2)$  gauge, but it would be possible to do a similar analysis of  $GL(2)$  invariants.

A clear limitation of the 4D scattering equations is that they currently only support tree-level calculations, and an obvious future direction is to investigate calculation of loop-level integrands using these methods. Loop-level scattering equations exist in general dimensions [50, 51], and future work could implement numerical algorithms to evaluate loop-level integrands as sums over solutions to the general  $d$  equations.

The 4D scattering equations are not restricted only to calculating amplitudes, they also cover form factors [52] and possibly further structures in quantum field theory. They can also be used to calculate more physically relevant standard model amplitudes [53]. The tools we have provided in this paper should be directly relevant to calculation of form factors, and could play a role in finding new types of structures that can be supported on scattering equations.

## Acknowledgments

I would like to thank Simon Badger, Jake Bourjaily, Paul Heslop and Dan Rutter for useful discussions, Tim Whitbread for help with numerical algorithms, Themis Botsas for guidance on statistical methods, Theresa Abl for bug testing the package, the community at [mathematica.stackexchange.com](https://mathematica.stackexchange.com) for help with many MATHEMATICA related questions, and especially Arthur Lipstein, without whose excellent supervision I wouldn't have been able to complete this work. I have been funded for this work by EPSRC PhD scholarship EP/L504762/1.

## A The MATHEMATICA package `treeamps4dJAF`

`treeamps4dJAF` provides a set of computational tools in MATHEMATICA for analytical and numerical calculation of amplitudes at tree-level. The package's most high-end functions are tailored for calculations in the 4D scattering equation formalism using the techniques derived in this paper. For example the package provides functionality for calculating MHV amplitudes analytically with specified external states in the supermultiplets of Yang-Mills and Einstein gravity theories, and graviton multiplets in conformal supergravity. Moving out of the MHV sector the package also provides functionality for solving the scattering equations numerically and using these solutions to calculate amplitudes. At a more low-level analysis of the code, many functions and a framework are provided for analytical computations in general in the spinner helicity formalism for amplitudes, including for example functions for dealing with antisymmetric brackets and the Schouten identity, and for evaluating expressions in terms of momentum twistors numerically. Each function is detailed in the documentation of the package, and can be accessed using `Information` via `?Amplitude` or `??Amplitude`. Here we provide an overview of the package's key functions.

The package reserves, clears and protects a number of default heads for usage with symbolic calculations. These can be accessed using `GetHeads` and `SetHeads`, and they are explained in table 3.

### A.1 Calculating amplitudes

The most high-level functions of `treeamps4dJAF` are used for calculation and readable display of analytical and numerical amplitudes.

`Amplitude` is the package's main function. It is used to calculate tree-level analytical MHV amplitudes and numerical amplitudes in all MHV sectors by solving the 4D scattering equations. Analytical output of `Amplitude` will be a function of the external momenta in terms angle and square brackets and Grassmann  $\eta$  variables; see table 3 for details. The input for `Amplitude` is overloaded in a number of different ways. Analytical amplitudes are specified in the form;

- `Amplitude[theory, N, states]`
- `Amplitude[theory, N, n, L]`

Name	Default Head	Mathematical Notation
"Angle"	ang	$\text{ang}[i,j] = \langle ij \rangle = \det( i\rangle  j\rangle)$ $\text{ang}[i,j,k,l] = \langle ijkl \rangle = \det(Z_i Z_j Z_k Z_l)$
"Square"	squ	$\text{squ}[i,j] = [ij] = \det( i\rangle  j\rangle)$
"Circle"	cir	$\text{cir}[i,j] = (ij) = \det(\sigma_i \sigma_j)$
"TwistorMu"	mu	$\text{mu}[i,j] = [\mu_i \mu_j] = \det(\mu_i \mu_j)$
"Lambda"	lam	$\text{lam}[i] =  i\rangle, \text{lam}[i,a] =  i\rangle^a$
"LambdaTilde"	lamt	$\text{lamt}[i] = [i], \text{lamt}[i,a] = [i]^a$
"Worldsheet"	s	$\text{s}[i,a] = \sigma_i^a$
"Grassmann"	eta	$\text{eta}[i,A] = \eta_i^A, \text{eta}[A] = \eta^A$

**Table 3.** The default heads reserved by `treeamps4dJAF` for analytical computations. See section 2 for definitions and conventions.

Numerical amplitudes can take any of the following forms;

- `Amplitude[theory, N, momenta, states]`
- `Amplitude[theory, N, momenta, n, L]`
- `Amplitude[theory, N, momenta, Lsolutions, solutions, states]`
- `Amplitude[theory, N, momenta, Lsolutions, solutions, L]`

$n$  and  $L$  specifications can be used to calculate Grassmann superamplitudes, and providing external states will perform the Grassmann integrations to give a component amplitude. If a set of numerical *momenta* matching the state specification are entered, `Amplitude` will attempt to solve the relevant scattering equations, and return either a numerical value or a superamplitude Grassmann expansion with numerical coefficients. On entering a set of numerical *momenta* along with left set  $L_{\text{solutions}}$  and a full set of solutions *solutions* which match the number of points and MHV degree of the specified states, `Amplitude` will sum the solutions over the relevant integrand and return the numerical amplitude.

In[1]:=	<code>Amplitude["EG",0,"h-","h-","h+"]</code>
Out[1]=	$\frac{\text{ang}[1,2]^6}{\text{ang}[1,3]^2 \text{ang}[2,3]^2}$
In[2]:=	<code>Amplitude["YM", 1, 3, {1, 2}]</code>
Out[2]=	$\frac{\text{ang}[1,2]^2 (-\text{ang}[1,2] \text{eta}[2,1] ** \text{eta}[1,1] - \text{ang}[1,3] \text{eta}[3,1] ** \text{eta}[1,1] - \text{ang}[2,3] \text{eta}[3,1] ** \text{eta}[2,1])}{\text{ang}[1,3] \text{ang}[2,3]}$
In[3]:=	<code>Amplitude["EG", 0, RandomMomenta4D[8], {1, 3, 5, 7}]</code>
Out[3]=	<code>11.8556 - 30.6627 I</code>

`AmplitudesDisplay[expr]` displays all of the package's heads (as explained in table 3) in a simplified format for readability.

```

In[1]:= Amplitude["YM",0,"g-","g-","g+","g+"]//AmplitudesDisplay
Out[1]= 
$$-\frac{\langle 12 \rangle^3}{\langle 23 \rangle \langle 34 \rangle \langle 14 \rangle}$$


In[2]:= Amplitude["YM",2,3,{1}] // AmplitudesDisplay
Out[2]= 
$$\frac{[23](\langle 23 \rangle \eta_1^1 + [12] \eta_3^1 - [13] \eta_2^1)(\langle 23 \rangle \eta_1^2 + [12] \eta_3^2 - [13] \eta_2^2)}{[12][13]}$$


```

`ListTheories[]` returns a list of all theories with number of supersymmetries currently supported by `treeamps4dJAF`.

```

In[1]:= ListTheories[]
Out[1]= {{ "YM", 0}, {"YM", 1}, {"YM", 2}, {"YM", 3}, {"YM", 4}, {"EG",
0}, {"EG", 1}, {"EG", 2}, {"EG", 3}, {"EG", 4}, {"EG", 5},
{"EG", 6}, {"EG", 7}, {"EG", 8}, {"CG", 4}}

```

`ListStates[theory, N]` returns a list of the specifications for all possible external states in *theory* with *N* supersymmetries, possibly with R symmetry indices.

```

In[1]:= ListStates["YM",2]
Out[1]= {"g-", {"Ψ-", A}, "Φ-", "Φ+", {"Ψ+", A}, "g+"}

```

## A.2 Solving the 4D scattering equations

The package's core functions for working with the 4D scattering equations are

`ScatteringEquations4D[n, L, gaugefix, delete]` returns the scattering equations at *n* points with left set *L*, with worldsheet variables specified by *gaugefix* fixed to the identity, and with equations given by *delete* removed. `ScatteringEquations4D[momenta, L, gaugefix, delete]` returns the equations as a function of worldsheet minors only, for the specified numerical *momenta*. If *gaugefix* and *delete* are not entered, `ScatteringEquations4D` returns the scattering equations at *n* points with left set *L* in terms of worldsheet minors.

```

In[1]:= ScatteringEquations4D[3, {1, 2}, {1, 2}]
Out[1]= {lam[3,1] -  $\frac{\text{lam}[2,1]}{s[3,1]}$  +  $\frac{\text{lam}[1,1]}{s[3,2]}$ , lam[3,2] -  $\frac{\text{lam}[2,2]}{s[3,1]}$  +  $\frac{\text{lam}[1,2]}{s[3,2]}$ }

In[2]:= ScatteringEquations4D[4, {1, 2}] // AmplitudesDisplay
Out[2]= {[1] -  $\frac{[3]}{(13)}$  -  $\frac{[4]}{(14)}$ , [2] -  $\frac{[3]}{(23)}$  -  $\frac{[14]}{(24)}$ , [3] -  $\frac{[1]}{(31)}$  -  $\frac{[2]}{(32)}$ , [4] -  $\frac{[1]}{(41)}$  -  $\frac{[2]}{(42)}$ }

```

`SolveScatteringEquations4D[n, L, gaugefix]` returns the analytical MHV and  $\overline{\text{MHV}}$  solutions to the scattering equations derived in section 3 at *n* points with left set *L* and particles specified by *gaugefix* gauge-fixed to the identity.

```

In[1]:= SolveScatteringEquations4D[4, {1, 2}]
Out[1]= {s[1,1] → 1, s[1,2] → 0, s[2,1] → 0, s[2,2] → 1, s[3,1] →  $\frac{\text{ang}[1,2]}{\text{ang}[1,3]}$ ,
s[3,2] →  $\frac{\text{ang}[1,2]}{\text{ang}[2,3]}$ , s[4,1] →  $\frac{\text{ang}[1,2]}{\text{ang}[1,4]}$ , s[4,2] →  $\frac{\text{ang}[1,2]}{\text{ang}[2,4]}$ }

```

`NSolveMonteCarlo[equations, variables, inputsolutions]` attempts to find numerical approximations to the solutions of the system *equations* for the *variables* by the Monte Carlo

algorithm outlined in section 4.2. *inputsols* is an optional argument for providing a list of previously known solutions to the equations. There are a number of different options for specifying details of the algorithm. *NumSolutions*, *MaxIterations* and *MaxTime* provide different stopping conditions, and *Distribution* sets the sampling distribution for initial points. Other options can be checked in the documentation. Calling *Abort* (Alt + .) during the calculation will return any solutions currently found.

```
In[1]:= NSolveMonteCarlo[{x^x + x^4-x - 4}, {x}, NumSolutions -> 5]
Out[1]= {{-0.864475 - 0.526651 I}, {1.39997}, {0.00678977 +
0.95836 I}, {0.00678977 - 0.95836 I}, {-0.864475 + 0.526651
I}}
```

*NSolveScatteringEquations4D*[*momenta*, *L*, *gaugefix*, *delete*, *inputsolutions*] attempts to find numerical approximations for all of the  $\binom{\text{Length}[\text{momenta}]-3}{\text{Length}[L]-2}$  solutions to the scattering equations defined by numerical *momenta* and left set *L* via the *NSolveMonteCarlo* algorithm. Optional argument *gaugefix* specified which worldsheet variables are gauge-fixed to the identity, and *delete* specifies which equations are removed. The initial random points are sampled from the distributions described in section 4.2. Any previously known solutions can be provided via *inputsolutions*.

```
In[1]:= NSolveScatteringEquations4D[RandomMomenta4D[4], {1, 2}]
Out[1]= {{{1, 0}, {0, 1}, {1.14552 - 1.31495 I, -1.08314 -
0.751286 I}, {0.191178 + 0.920906 I, -0.461597 + 0.613013
I}}}
```

### A.3 Applying momenta and solutions to expressions

*treeamps4DJAF* provides a number of functions for evaluating analytical expressions on specific numerical external data.

*RandomMomenta4D*[*n*, *w*] samples a set of complex numbers with rational real and imaginary parts in the interval  $[-w, w]$ , and returns *n* point complexified momenta satisfying momentum conservation. Each element is in the form  $\{|i\rangle, [i]\}$ , and momentum conservation is solved by setting  $|1\rangle = \sum_{i=3}^n \frac{\langle 2i \rangle}{\langle 12 \rangle} |i\rangle$  and  $|2\rangle = \sum_{i=3}^n \frac{\langle 1i \rangle}{\langle 21 \rangle} |i\rangle$ .

*ApplyMomenta*[*expression*, *momenta*] replaces the package's heads for spinor helicity variables with the *momenta* in the given *expression*.

```
In[1]:= ApplyMomenta[ang[1, 2] lam[3], RandomMomenta4D[6]]
Out[1]= {6141/250000 - (49771 I)/1000000, 4071641249/125000000000 +
(22520160139 I)/1000000000000}
```

*Integrand* returns a theory-dependent tree-level integrand which can be summed over solutions to the scattering equations to return an amplitude. The output of *Integrand* will be an analytical expression which is a function of the worldsheet and external momenta,



and Grassmann  $\eta$  variables. The input for `Integrand` is overloaded in different ways, for example;

- `Integrand[theory, N, states]`
- `Integrand[theory, N, n, L]`

The specifications for the states work the same as for `Amplitude`, which inherits its specifications from `Integrand`.

```
In[1]:= Integrand["EG", 0, 4, {1, 2}]
Out[1]=  $\frac{\text{ang}[1,2]\text{squ}[3,4]}{\text{cir}[1,2]\text{cir}[3,4]}$ 

In[2]:= Integrand["CG", 4, "h-", "h-", "h+", "h+"]
Out[2]=  $\frac{\text{ang}[1,2]^2\text{squ}[3,4]^2}{\text{cir}[1,2]^2\text{cir}[3,4]^2}$ 
```

`ApplyScatteringSolutions[expression, momenta, L, solutions]` applies numerical *momenta* to *expression*, and then sums *expression* and the relevant Jacobian over the given solutions to the scattering equations, for the number of points specified by the length of *momenta* and left set *L*.

```
In[1]:= mom = RandomMomenta4D[6];
        sols = NSolveScatteringEquations4D[mom, {1, 2, 3}];
        ApplyScatteringSolutions[cir[1,2]cir[3,4]cir[5,6], mom,
        {1, 2, 3}, sols]
Out[1]= -23.1883 + 27.2079 I
```

#### A.4 Grassmann integration and swapping left set algorithms

`DetGIntegrate[expr, measure, space]` implements the algorithm from section 4.3 for efficient Grassmann integration.

```
In[1]:= DetGIntegrate[(a eta[1] + b eta[2]) ** (c eta [1] + d
        eta[2]), {{1}, {2}}, {{1}, {2}}]
Out[1]= -b c + a d
```

`GIntegrateStates[theory, N, expression, states]` selects the relevant Grassmann  $\eta$  variables associated with the *states* for *theory* with *N* supersymmetries, and uses these as the measure to integrate *expression*.

```
In[1]:= GIntegrateStates["YM", 4, MHVBarGrassmannFactor[4, 3], "g-",
        "g+", "g+"]
Out[1]=  $\text{squ}[2, 3]^4$ 
```

`SwapParticles[integrand, n, L, l0, r0]` applies the worldsheet transformation specified in section B.4 to the *n* point *integrand* with left set *L*, swapping *l0* and *r0* in between the left set and the right set.



`ChangeLeftSet`[*integrand*, *n*, *Lold*, *Lnew*] applies `SwapParticles` repeatedly until the *integrand* supported on *Lold* is supported on *Lnew*

```
In[1]:= ChangeLeftSet[1/(cir[1, 2] cir[2, 3] cir[3, 1]), 3, {1, 2},
           {1, 3}]
Out[1]= - 1/cir[1,2]cir[1,3]cir[2,3]^5
```

## A.5 File input-output for sets of solutions

Solving the 4D scattering equations using `NSolveScatteringEquations4D` is stochastic, and can take a long time for higher points and MHV degrees. `treeamps4dJAF` provides functions for storing and reading set of Solutions to and from a CSV file, along with a lookup table of solutions in `SolutionLookupTable.csv` which can be used without solving the equations.

`ReadSolution`[*filename*, *n*, *L*, *solutionnumber*] looks for a solution specified by *n* and *L* in *filename.csv*, and returns the relevant data if found.

```
In[1]:= ReadSolution["SolutionLookupTable", 4, {1, 2}]
Out[1]= {{{{1, 0}, {0, 1}, {1.14552 - 1.31495 I, -1.08314 - 0.751286
           I}, {0.191178 + 0.920906 I, -0.461597 + 0.613013 I}}}}
```

Importing large CSV files directly into MATHEMATICA can be very slow.

`ReadSolutionData`[*filename*] opens *filename.csv* and returns the data for each row of *filename.csv* in the form {number of points, left set, gauge-fixed rows} without actually loading the solutions or momenta into the RAM.

`AppendSolution`[*filename*, *momenta*, *L*, *solutions*] opens *filename.csv* and appends at the end of it a new row containing the data {Length[*momenta*], *L*, GetGaugeFix[solutions], *momenta*, *solutions*}, with the gauge-fixed columns removed from *solutions*.

`SolveAppendSolution`[*filename*, *momenta*, *L*, *inputsolutions*] attempts to find numerical solutions to the scattering equations and then append them to *filename.csv*.

## B Analytical details of the 4D scattering equations

In this appendix we provide detailed proofs of the analytical results from section 3, along with some further general results for the 4D scattering equations.

### B.1 Recovering the general $d$ scattering equations for $d = 4$

Solutions to the 4D equations are grouped into sets for the different  $N^{k-2}$ MHV sectors. The general  $d$  equations depend only on  $n$  and hence do not encode this grouping. As the 4D scattering equations contain more information than the general  $d$  scattering equations we have  $4D \Rightarrow$  general  $d$  (for  $d = 4$ ), which we prove in this section. A different argument

is given in [37]. The proof also provides an explicit method for finding solutions to the general  $d$  equations using those from the 4D specific case. Integrands for the general  $d$  equations for  $d = 4$  can also be mapped to integrands for the 4D equations [54].

First we prove a lemma which holds for the general  $d$  equations. Define a world-sheet dependent momentum  $P(s) := \sum_{j \in N} \frac{k_i}{s - s_i}$ . Then we prove that  $P(s)^2 = 0 \Leftrightarrow$  the general  $d$  equations. Note that there are no second order poles in  $P(s)^2$  as all of the external momenta  $k_i$  are null.

$$P(s)^2 = \sum_{\substack{i, j \in \mathcal{N} \\ i \neq j}} \frac{k_i \cdot k_j}{(s - s_i)(s - s_j)} = 2 \sum_{i \in \mathcal{N}} \frac{1}{s - s_i} \left( \sum_{\substack{j \in \mathcal{N} \\ j \neq i}} \frac{k_i \cdot k_j}{s_i - s_j} \right), \quad (\text{B.1})$$

where we use partial fractions and relabel indices to arrive at the result.  $P(s)$  is now written explicitly as a sum of its poles, and hence can only be zero if all of its residues are zero. Then we conclude that  $P(s)^2 = 0 \Leftrightarrow \sum_{j \in \mathcal{N}} \frac{k_i \cdot k_j}{s_i - s_j} = 0$ .

To relate to the 4D equations, construct a new explicitly null world-sheet dependent momentum in terms of two world-sheet dependent spinors  $|\lambda(s)\rangle := \sum_{r \in R} \frac{t_r |r\rangle}{s - s_r}$  and  $|\lambda(s)] := \sum_{l \in L} \frac{t_l |l]}{s - s_l}$ .

$$\begin{aligned} |\lambda(s)] \langle \lambda(s)| &= \left( \sum_{r \in R} \frac{t_r |r\rangle}{s - s_r} \right) \left( \sum_{l \in L} \frac{t_l \langle l|}{s - s_l} \right) = \sum_{\substack{r \in R \\ l \in L}} \frac{|r\rangle \langle l|}{(rl)} \left( \frac{1}{s - s_l} - \frac{1}{s - s_r} \right) \\ &= \sum_{r \in R} \frac{|r\rangle}{s - s_r} \left( \sum_{l \in L} \frac{\langle l|}{(rl)} \right) + \sum_{l \in L} \left( \sum_{r \in R} \frac{|r\rangle}{(lr)} \right) \frac{\langle l|}{s - s_l} \\ &= \sum_{r \in R} \frac{|r\rangle \langle r|}{s - s_r} + \sum_{l \in L} \frac{|l\rangle \langle l|}{s - s_l} = P(s), \end{aligned} \quad (\text{B.2})$$

where we use the same steps as in the previous calculation, and in the second last equality we use the 4D scattering equations and  $k_i = |i\rangle \langle i|$ . Given that  $|\lambda(s)] \langle \lambda(s)|$  is explicitly constructed as a null vector, we then see that the 4D scattering equations imply that  $P(s)^2 = 0$ , and hence by the lemma they imply the general  $d$  scattering equations.

This proof also provides an explicit mapping from a solution to the 4D scattering equations to a solution to the general  $d$  equations. We map a point in the solution space of the 4D equations to a point in the  $n$ -punctured Riemann sphere by writing each column as  $\sigma_i = t_i^{-1} \begin{pmatrix} 1 \\ s_i \end{pmatrix}$ , and keeping only the  $s$  variables and not the scales  $t$ . Suppose that we have some solution to the 4D scattering equations with the gauge-fixing that the first two particles are equal to the identity matrix. Under this mapping the vector  $\begin{pmatrix} 1 \\ 0 \end{pmatrix}$  maps to the point at infinity, and we fix the remaining gauge redundancy by dividing through by  $s_3$  to arrive at

$$\sigma_{4D} = \begin{pmatrix} 1 & 0 & \sigma_3^1 & \dots & \sigma_n^1 \\ 0 & 1 & \sigma_3^2 & \dots & \sigma_n^2 \end{pmatrix} \rightarrow s_{\text{general } d, d=4} = \left( \infty \ 0 \ 1 \ \frac{\sigma_4^1}{\sigma_4^2} \frac{\sigma_3^2}{\sigma_3^1} \dots \frac{\sigma_n^1}{\sigma_n^2} \frac{\sigma_3^2}{\sigma_3^1} \right). \quad (\text{B.3})$$

From this analysis we see that reconstructing a full solution to the 4D equations in terms of the  $s$  and  $t$  variables is not direct given a  $d = 4$  solution to the general  $d$  equations, and it would be interesting to understand how the  $t$  variables can be specified in this case.

MHV and  $\overline{\text{MHV}}$  solutions to the general  $d$  equations for  $d = 4$  were derived in [45]. Using the mapping (B.3), we can see explicitly how the MHV solutions to the 4D equations derived in section 3 map onto those of the general  $d$  equations

$$\sigma_{\text{MHV}} = \begin{pmatrix} 1 & 0 & \frac{\langle 12 \rangle}{\langle 31 \rangle} & \cdots & \frac{\langle 12 \rangle}{\langle n1 \rangle} \\ 0 & 1 & \frac{\langle 12 \rangle}{\langle 32 \rangle} & \cdots & \frac{\langle 12 \rangle}{\langle n2 \rangle} \end{pmatrix} \rightarrow s_{\text{MHV}} = \begin{pmatrix} \infty & 0 & 1 & \frac{\langle 41 \rangle \langle 32 \rangle}{\langle 31 \rangle \langle 42 \rangle} & \cdots & \frac{\langle n1 \rangle \langle 32 \rangle}{\langle 31 \rangle \langle n2 \rangle} \end{pmatrix}, \quad (\text{B.4})$$

in agreement with equation (49) of [45], up to choice of  $\text{SL}(2)$  fixing.

## B.2 Symmetries, little group scaling and Grassmanians

The scattering equations have a  $\text{GL}(2)$  symmetry which can be realised in different ways in terms of a worldsheet redefinition, or a worldsheet redefinition with a corresponding little group rescaling. The worldsheet  $\text{GL}(2)$  symmetry in (3.1) is a combination of the standard  $\text{SL}(2)$  symmetry of global conformal transformations in the string worldsheet  $s$  variables, and a  $\text{GL}(1)$  transformation corresponding to a rescaling of the worldsheet  $t$  variables. Any function  $f(\sigma)$  which is integrated against the scattering equation delta functions must transform as  $f(\sigma) \rightarrow f(\sigma)(\det G)^{n-2k}$  under (3.1) to balance out the transformation of the measure. All of the integrands for the theories considered in section 2 satisfy this transformation law, as enforced by their underlying 4D ambitwistor string models [21].

Before considering joint worldsheet and little group transformations, we first analyse the little group scaling of amplitudes supported on the scattering equations. First consider a general amplitude  $\mathcal{A}_{n,L}$  with some arbitrary integrand  $f(\sigma, |i\rangle, |i])$ ,

$$\mathcal{A}_{n,L} := \int \frac{d^{2 \times n} \sigma}{\text{GL}(2)} \delta^{2 \times n} (SE_L^n) f(\sigma, |i\rangle, |i]).$$

Perform a different little group scaling for each particle, such that  $|i\rangle \rightarrow \alpha_i |i\rangle$ ,  $|i] \rightarrow \alpha_i^{-1} |i]$ . Then we see that

$$\mathcal{A}_{n,L}(\alpha_i |i\rangle, \alpha_i^{-1} |i]) = \int \frac{d^{2 \times n} \sigma}{\text{GL}(2)} \prod_{l \in L} \delta^2 \left( \frac{|l]}{\alpha_l} - \sum_{r \in R} \frac{|r]}{\alpha_r(lr)} \right) \prod_{r \in R} \delta^2 \left( \alpha_r |r\rangle - \sum_{l \in L} \frac{\alpha_l |l\rangle}{(rl)} \right) f(\sigma, \alpha_i |i\rangle, \alpha_i^{-1} |i]). \quad (\text{B.5})$$

Define new worldsheet coordinates such that  $\sigma'_l := \alpha_l^{-1} \sigma_l$  for  $l \in L$ , and  $\sigma'_r := \alpha_r \sigma_r$  for  $r \in R$ . Change variables and rename back to  $\sigma$ , picking up factors of the  $\alpha_i$  from the delta functions and the measure to arrive at

$$\mathcal{A}_{n,L}(\alpha_i |i\rangle, \alpha_i^{-1} |i]) = \int \frac{d^{2 \times n} \sigma}{\text{GL}(2)} \delta^{2 \times n} (SE_L^n) \prod_{l \in L} \alpha_l^4 \prod_{r \in R} \alpha_r^{-4} f(\alpha_l \sigma_l, \alpha_r^{-1} \sigma_r, \alpha_i |i\rangle, \alpha_i^{-1} |i]). \quad (\text{B.6})$$

Now consider how the integrand for Yang-Mills theory with  $\mathcal{N}$  supersymmetries as defined in section 2 scales under the little group. Under this little group transformation the Grassmann variables transform as  $\eta_i \rightarrow \alpha_i^{-1} \eta_i$ , and the Grassmann delta functions

transform in a similar way to the scattering equation delta functions. The transformation of the Parke-Taylor factor cancels out that of the measure, and the integrand becomes

$$f_{\text{SYM}}(\alpha_l \sigma_l, \alpha_r^{-1} \sigma_r, \alpha_i |i\rangle, \alpha_i^{-1} |i], \alpha_i^{-1} \eta_i) = \prod_{l \in L} \alpha_l^{-2-\mathcal{N}} \prod_{r \in R} \alpha_r^2 f_{\text{SYM}}(\sigma, |i\rangle, |i], \eta_i). \quad (\text{B.7})$$

From this we find the standard little group scaling for negative and positive helicity superfields of a Yang-Mills superamplitude, that

$$\mathcal{A}_{n,L}(\alpha_i |i\rangle, \alpha_i^{-1} |i], \alpha_i^{-1} \eta_i) = \prod_{l \in L} \alpha_l^{2-\mathcal{N}} \prod_{r \in R} \alpha_r^{-2} \mathcal{A}_{n,L}(|i\rangle, |i], \eta_i). \quad (\text{B.8})$$

A similar analysis produces the required scaling for gravity amplitudes.

Combining these two types of transformations we find a symmetry which acts with a standard  $\text{GL}(2)$  transformation such that  $\sigma \rightarrow G\sigma$ , and hence the minors of the  $\sigma$  matrix transform as  $(ij) \rightarrow \det G (ij)$ . Simultaneously performing an inhomogeneous little group scaling such that  $|l] \rightarrow \alpha |l]$ ,  $|l\rangle \rightarrow \alpha^{-1} |l\rangle$  for  $l \in L$  and  $|r] \rightarrow \beta |r]$ ,  $|r\rangle \rightarrow \beta^{-1} |r\rangle$  for  $r \in R$ , the scattering equations become

$$\begin{aligned} \delta^{2 \times n}(SE_L^n) &= \prod_{l \in L} \alpha^{-2} \delta^2 \left( |l] - \frac{\beta}{\alpha \det G} \sum_{r \in R} \frac{|r]}{(lr)} \right) \prod_{r \in R} \beta^2 \delta^2 \left( |r\rangle - \frac{\beta}{\alpha \det G} \sum_{l \in L} \frac{|l\rangle}{(rl)} \right) \\ &= \alpha^{2n-4k} (\det G)^{2n-2k} \delta^{2 \times n}(SE_L^n), \end{aligned} \quad (\text{B.9})$$

where in the last equation we choose  $\beta = \alpha \det G$  to keep the equations invariant. The measure and delta functions combined transform such that

$$\begin{aligned} \int \frac{d^{2 \times n} \sigma}{\text{GL}(2)} \delta^{2 \times n}(SE_L^n) f(\sigma, |i\rangle, |i]) &\rightarrow \int \frac{d^{2 \times n} \sigma}{\text{GL}(2)} \delta^{2 \times n}(SE_L^n) (\det G)^{3n-2k} \alpha^{2n-4k} \\ &\quad f(G\sigma, \alpha |l\rangle, \alpha^{-1} |l], \alpha \det G |r\rangle, (\alpha \det G)^{-1} |r]). \end{aligned} \quad (\text{B.10})$$

We have seen how the little group transformation of the amplitude  $\mathcal{A}$  comes from little group covariance of the integrand  $f$ . Any  $f$  which integrates to an amplitude must transform covariantly under the little group, and hence we need that  $f$  transforms as a scaling transformation for any combined little group transformation and worldsheet rescaling. We can then conclude that for any  $f$  which describes an amplitude there must exist some  $x$  and  $y$  real numbers such that

$$f(G\sigma, \alpha |l\rangle, \alpha^{-1} |l], \alpha \det G |r\rangle, (\alpha \det G)^{-1} |r]) = (\det G)^x \alpha^y f(\sigma, |i\rangle, |i]).$$

Given  $f$  which transforms in this way, we can choose the little group scaling  $\alpha$  to such that  $(\det G)^{3n-2k+x} \alpha^{2n-4k+y} = 1$ , and hence this transformation is a symmetry for any amplitude supported on the scattering equations. This  $\text{GL}(2)$  invariance is the  $\text{GL}(2)$  invariance of the Grassmannian  $\text{Gr}(2, n)$ , and hence in this sense we can think of the solutions to the scattering equations as living in  $\text{Gr}(2, n)$ .

### B.3 Deleting equations and momentum conservation

In this appendix we demonstrate how to remove four equations to give a momentum conserving delta function. There are three possible cases; we either consider the equations for two particles in the left set, or for two in the right set, or for one in each. First consider the case of two particles in the left set. Without loss of generality, label these particles to be 1 and 2. Defining the delta functions for these particles to be  $\Delta_{1,2}$  we see that

$$\begin{aligned}\Delta_{1,2} &:= \delta^2 \left( |1\rangle - \sum_{r \in R} \frac{|r\rangle}{(1r)} \right) \delta^2 \left( |2\rangle - \sum_{r \in R} \frac{|r\rangle}{(2r)} \right) \\ &= \langle 12 \rangle^4 \delta^2 \left( |1\rangle \langle 12| - \sum_{r \in R} \frac{|r\rangle \langle 12|}{(1r)} \right) \delta^2 \left( |2\rangle \langle 21| - \sum_{r \in R} \frac{|r\rangle \langle 21|}{(2r)} \right).\end{aligned}\tag{B.11}$$

Now consider a general right set equation and contract with first with  $\langle 2|$ , and separately with  $\langle 1|$ , to find that

$$\begin{aligned}|r\rangle = \sum_{l \in L} \frac{|l\rangle}{(rl)} &\implies \frac{\langle 12 \rangle}{(1r)} = -\langle r2 \rangle - \sum_{\substack{l \in L \\ l \neq 1,2}} \frac{\langle l2 \rangle}{(lr)} \\ \frac{\langle 21 \rangle}{(2r)} &= -\langle r1 \rangle - \sum_{\substack{l \in L \\ l \neq 1,2}} \frac{\langle l1 \rangle}{(lr)}.\end{aligned}\tag{B.12}$$

Substituting into  $\Delta_{1,2}$ , we see that

$$\begin{aligned}\Delta_{1,2} &= \langle 12 \rangle^4 \delta^2 \left( \left( |1\rangle \langle 1| + \sum_{r \in R} |r\rangle \langle r| \right) |2\rangle + \sum_{\substack{r \in R, l \in L \\ l \neq 1,2}} \frac{|r\rangle \langle l2|}{(lr)} \right) \\ &\quad \delta^2 \left( \left( |2\rangle \langle 2| + \sum_{r \in R} |r\rangle \langle r| \right) |1\rangle + \sum_{\substack{r \in R, l \in L \\ l \neq 1,2}} \frac{|r\rangle \langle l1|}{(lr)} \right).\end{aligned}\tag{B.13}$$

We then use the remaining left set equations to solve the sum over  $r$  in the last term in each delta function, arriving at

$$\Delta_{1,2} = \langle 12 \rangle^4 \delta^2 \left( \left( \sum_{n \in \mathcal{N}} |n\rangle \langle n| \right) |2\rangle \right) \delta^2 \left( \left( \sum_{n \in \mathcal{N}} |n\rangle \langle n| \right) |1\rangle \right) = \langle 12 \rangle^2 \delta^4(P).\tag{B.14}$$

We would now say that we have ‘deleted equations 1 and 2’, and the remaining  $2n - 4$  equations now give a well-specified system. Note the Jacobian  $\langle 12 \rangle^2$  for this calculation. The calculation for deleting two equations in the right set goes by the same steps, and labelling the two particles in the right set to be  $r_1$  and  $r_2$  we find that

$$\delta^2 \left( |r_1\rangle - \sum_{l \in L} \frac{|l\rangle}{(r_1 l)} \right) \delta^2 \left( |r_2\rangle - \sum_{l \in L} \frac{|l\rangle}{(r_2 l)} \right) = [r_1 r_2]^2 \delta^4(P).\tag{B.15}$$

There is one remaining choice; we could delete one equation from the left set and one equation from the right set. Choosing equations in this way does not produce a momentum-conservation delta function, and hence does not result in a solvable system for spinors satisfying momentum conservation. Label the left set particle as  $1 \in L$ , and the right set particle as  $n \in R$ . Then following the same analysis as above, we arrive at

$$\begin{aligned} \delta^2 \left( |1\rangle - \sum_{r \in R} \frac{|r\rangle}{(1r)} \right) \delta^2 \left( |n\rangle - \sum_{l \in L} \frac{|l\rangle}{(nl)} \right) \\ = \langle 1n \rangle^2 [1n]^2 \delta^2 \left( |1\rangle \langle 1n\rangle + \sum_{i=2}^{n-1} |i\rangle \langle in\rangle \right) \delta^2 \left( \sum_{i=2}^{n-1} [1i] \langle i| + [1n] \langle n| \right). \end{aligned} \quad (\text{B.16})$$

These equations look deceptively similar to momentum conservation. We prove here that they are in fact not the same, and give a constraint corresponding to non-generic kinematics.

Solving the first equation as  $|1\rangle = \frac{1}{\langle n1\rangle} \sum_{i=2}^{n-1} |i\rangle \langle in\rangle$  and substituting into the second, we find that these delta functions imply that

$$|n\rangle \sum_{i=2}^{n-1} P_n \cdot P_i + \sum_{\substack{i,j=2 \\ j \neq i}}^{n-1} |i\rangle [ij] \langle jn\rangle = |n\rangle \left( \sum_{i=2}^{n-1} P_i \right)^2 = 0, \quad (\text{B.17})$$

where between the first two equalities we split the second sum into two terms, relabel the indices use a Schouten identity. Hence to keep consistency with these equations we must set either  $|n\rangle$  or the Mandelstam invariant  $\left( \sum_{i=2}^{n-1} P_i \right)^2$  to zero, and neither of these choices correspond to generic kinematics. From this we see that it is not possible to delete one equation from each set.

#### B.4 Permutations and choice of left set

A given  $n$  point  $N^{k-2}$ MHV amplitude will be supported on scattering equations with a specified left set  $L$ . In this section we show how solutions to the scattering equations for one left set can be used to calculate amplitudes with the same MHV degree that are supported on different left set  $L'$ . This mapping of different left sets is important computationally as it will allow us to solve the equations only once in each MHV sector and then calculate all amplitudes of this MHV degree for the specified numerical momenta. The explicit worldsheet transformation swapping particles  $l_0 \in L$  and  $r_0 \in R$  in between the sets is given in equation (3.6). Under this transformation, we find that the scattering equations transform as

$$\begin{aligned} E_l &\longrightarrow E_l - \frac{(l_0 r_0)}{(l r_0)} E_{l_0} & E_{l_0} &\longrightarrow (l_0 r_0) E_{r_0} \\ E_r &\longrightarrow E_r - \frac{(r_0 l_0)}{(r l_0)} E_{r_0} & E_{r_0} &\longrightarrow (l_0 r_0) E_{l_0}, \end{aligned}$$

where we used a Schouten identity in the worldsheet variables for the  $l$  and  $r$  equations. The second terms in the  $l$  and  $r$  equations are zero on support of the  $l_0$  and  $r_0$  equations, and

hence we see that the scattering equations remain the same up to changing the particles  $l_0$  and  $r_0$  between the left and right set. The delta functions pick up an overall factor of  $(l_0 r_0)^{-4}$ . Calculating the transformation of the measure, we find that the scattering equation integral transforms as in equation (3.7).

If we now also perform a permutation of the external data on the same legs  $l_0$  and  $r_0$  and look at the transformation properties of the integrands, we can understand how the amplitude does not depend on a choice of left set for maximally supersymmetric theories and for  $\mathcal{N} = 4$  conformal supergravity. This transformation can also be used to understand permutation invariance under swapping between the right and left sets for  $\mathcal{N} = 8$  supergravity. Showing permutation invariance of gravity amplitudes under a swapping two legs which carry the same helicity superfield is straightforward, and simply requires renaming the worldsheet variables on the permuted legs.

Let us now look at the form of the integrand for Yang-Mills theories. Under this transformation, we find that

$$\begin{aligned} \int \frac{d^{2 \times n} \sigma}{\text{GL}(2)} \delta^{2 \times n}(SE_L^n) \frac{\prod_{l \in L} \delta^{\mathcal{N}} \left( \eta_l - \sum_{r \in R} \frac{\eta_r}{(lr)} \right)}{\prod_{i \in N} (i \ i+1)} \\ \rightarrow \int \frac{d^{2 \times n} \sigma}{\text{GL}(2)} \delta^{2 \times n}(SE_{L'}^n) \frac{\prod_{l \in L'} \delta^{\mathcal{N}} \left( \eta_l - \sum_{r \in R'} \frac{\eta_r}{(lr)} \right)}{\prod_{i \in N} (i \ i+1)} (l_0 r_0)^{\mathcal{N}-4}. \end{aligned} \quad (\text{B.18})$$

For  $\mathcal{N} = 4$  the integrand is unchanged up to a choice of the left set, and hence for  $\mathcal{N} = 4$  super Yang-Mills theory we can see that the choice of left set does not affect the overall superamplitude. Now look at the MHV sector for  $\mathcal{N} = 0$ , fix  $L = \{1, 2\}$  and swap the external data for particles 1 and  $r_0$ , as well as performing the integral transformation to swap 1 and  $r_0$ . We see that  $(1r_0)^4 = \frac{\langle 12 \rangle^4}{\langle 2r_0 \rangle^4}$ , which is exactly the factor required to modify the Park-Taylor formula for particles 1 and 2 negative helicity gluons to have particles 2 and  $r_0$  with negative helicities. It is interesting to note that this structure extends outside of the MHV sector at the level of the integrand in the scattering equation formalism.

## B.5 The Jacobian of the scattering equations

In this appendix we detail some properties of the Jacobian of the 4D scattering equations. For a general worldsheet integral over the scattering equation delta functions, we can solve the integrals and write the expression as a sum over the solutions to the scattering equations. To do this we need an explicit expression for the Jacobian  $J_L^n(\sigma)$  as follows. We use the notation for matrix  $A$  that  $A^{ij}$  has rows and columns  $i$  and  $j$  removed. Let  $l, l' \in L$  and delete equations  $l$  and  $l'$  to arrive at

$$\int \frac{d^{2 \times n} \sigma}{\text{GL}(2)} \delta^{2 \times n}(SE_L^n) f(\sigma) = \delta^4(P) \sum_{\sigma_{\text{sol}} \in \text{solutions}} \frac{f(\sigma_{\text{sol}})}{\langle ll' \rangle^{-2} \det(J_L^n{}^{ll'}(\sigma_{\text{sol}}))}. \quad (\text{B.19})$$

Note that as shown in appendix B.3, the two rows/columns deleted must either both be in the left set or both in the right set, and there is an extra associated factor eg.  $\langle ll' \rangle^2$  such that the full determinant of the Jacobian is  $\langle ll' \rangle^{-2} \det(J_L^n{}^{ll'})$  for two left set particles deleted and gauge-fixed.

At  $n$  points with left set  $L$  we find the Jacobian to be

$$J_L^n = \begin{pmatrix} \frac{\partial \tilde{E}_l}{\partial \sigma_{l'}} & \frac{\partial \tilde{E}_l}{\partial \sigma_{r'}} \\ \frac{\partial E_r}{\partial \sigma_{l'}} & \frac{\partial E_r}{\partial \sigma_{r'}} \end{pmatrix} = \begin{pmatrix} -\delta_{ll'} \sum_{r \in R} \frac{|r] \otimes \sigma_r}{(lr)^2} & \frac{|r'] \otimes \sigma_l}{(lr')^2} \\ \frac{\langle l| \otimes \sigma_r}{(lr)^2} & -\delta_{rr'} \sum_{l \in L} \frac{\langle l| \otimes \sigma_l}{(lr)^2} \end{pmatrix}, \quad (\text{B.20})$$

where the matrix is written in a block form with blocks of sizes of the left and right set, and each element of these matrices is broken down into a  $2 \times 2$  matrix which is a tensor product of a spinor with a worldsheet vector. This matrix has determinant 0, which is simple to check analytically for example in MATHEMATICA. This is insured by the fact that gauge-fixing removes 4 of the sigma variables, and hence we have to remove four of the rows of the matrix to produce a well specified system. Similarly we must remove four of the columns of  $J_L^n$ , which is equivalent to deleting two equations as shown in appendix B.3.

In the MHV sector this matrix is block diagonal and we can calculate the determinant in terms worldsheet minors directly. Taking the left set to the particles 1 and 2 and also gauge-fixing and deleting these particles we find that

$$\det \left( J_{\{1,2\}}^n{}^{12} \right) = \det \left( \frac{\partial E_r}{\partial \sigma_{r'}} \right) = \prod_{r \in R} \det \left( \frac{\langle 1| \otimes \sigma_1}{(1r)^2} + \frac{\langle 2| \otimes \sigma_2}{(2r)^2} \right). \quad (\text{B.21})$$

We now use the general result for determinants of sums of tensor products of two dimensional vectors that

$$\det \left( \sum_{i=1}^m \alpha_i u_i \otimes v_i \right) = \sum_{1 \leq i < j \leq m} \alpha_i \alpha_j \det(u_i u_j) \det(v_i v_j), \quad (\text{B.22})$$

for  $m$  variables  $u_i, v_i \in \mathbb{C}^2$  and  $\alpha_i \in \mathbb{C}$ . For the MHV calculation  $m = 2$ , giving

$$\det \left( J_{\{1,2\}}^n{}^{12} \right) = \prod_{r \in R} \frac{\langle 12 \rangle (12)}{(1r)^2 (2r)^2} = \prod_{r \in R} \frac{\langle 1r \rangle^2 \langle 2r \rangle^2}{\langle 12 \rangle^3}, \quad (\text{B.23})$$

where in the last equation we substitute in the MHV solution from section 3.

It is also possible to analytically evaluate the Jacobian outside of the MHV sector. Assume  $1, 2 \in L$  and gauge-fix and delete particles 1 and 2, and use the formula for the determinant of a block matrix to find that

$$\det \left( J_L^n{}^{12} \right) = \det \left( \frac{\partial E_r}{\partial \sigma_{r'}} \right) \det \left( \frac{\partial \tilde{E}_l}{\partial \sigma_{l'}} - \frac{\partial \tilde{E}_l}{\partial \sigma_{r'}} \left( \frac{\partial E_r}{\partial \sigma_{r'}} \right)^{-1} \frac{\partial E_r}{\partial \sigma_{l'}} \right), \quad (\text{B.24})$$

where  $r, r' \in R$  and  $l, l' \in L/\{1, 2\}$ .

As  $\frac{\partial E_r}{\partial \sigma_{r'}}$  is block diagonal, it is comparatively simple to calculate its determinant and inverse. Using equation (B.22) we find the determinant to be

$$\det \left( \frac{\partial E_r}{\partial \sigma_{r'}} \right) = \prod_{r \in R} \sum_{l < l' \in L} \frac{\langle ll' \rangle (ll')}{(rl)^2 (rl')^2}. \quad (\text{B.25})$$

To invert this matrix we see that in the  $r, r'$  indices it is simply  $\delta_{rr'}$ , leaving the calculation of the inverses of the  $2 \times 2$  blocks. We use the result for a  $2 \times 2$  matrix  $M$  that  $M^{-1} =$



$\det(M)^{-1} \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} M \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$  to invert in the spinor and worldsheet indices. This corresponds to raising and lowering the two indices, and dividing by the determinant as calculated by equation (B.22). The result is then that

$$\det \left( \frac{\partial \tilde{E}_l}{\partial \sigma_{l'}} - \frac{\partial \tilde{E}_l}{\partial \sigma_{r'}} \left( \frac{\partial E_r}{\partial \sigma_{r'}} \right)^{-1} \frac{\partial E_r}{\partial \sigma_{l'}} \right) \quad (\text{B.26})$$

$$= \det \left( \sum_{r \in R} \frac{[r] \otimes \sigma_r}{(lr)^2 \sum_{\lambda < \lambda' \in L} \frac{\langle \lambda \lambda' \rangle \langle \lambda \lambda' \rangle}{(r\lambda)^2 (r\lambda')^2}} \left( \sum_{\lambda \in L} \frac{\langle l' \lambda \rangle \langle l \lambda \rangle}{(\lambda r)^2 (l' r)^2} - \delta_{ll'} \sum_{\lambda < \lambda' \in L} \frac{\langle \lambda \lambda' \rangle \langle \lambda \lambda' \rangle}{(\lambda r)^2 (\lambda' r)^2} \right) \right),$$

where the determinant is taken over  $l, l' \in L/\{1, 2\}$ , combined with the tensor product of spinor and worldsheet indices. In the NMHV sector we see that  $l = l' = 3$  and we can calculate the determinant of the remaining  $2 \times 2$  matrix using equation (B.22). As an example at 6 points NMHV we find that

$$\det(J_{\{1,2,3\}}^6)^{12} = \frac{\langle 12 \rangle^2 \langle 12 \rangle^2}{\prod_{l,r} (lr)^2} \sum_{\substack{r < r' \in R \\ l < l' \in L}} [rr'] (rr') \langle ll' \rangle (ll') \sum_{\substack{\lambda \in L \\ \rho \in R}} \epsilon_{ll'\lambda} \epsilon_{rr'\rho} (\lambda \rho)^2. \quad (\text{B.27})$$

**Open Access.** This article is distributed under the terms of the Creative Commons Attribution License ([CC-BY 4.0](https://creativecommons.org/licenses/by/4.0/)), which permits any use, distribution and reproduction in any medium, provided the original author(s) and source are credited.

## References

- [1] Z. Xu, D.-H. Zhang and L. Chang, *Helicity Amplitudes for Multiple Bremsstrahlung in Massless Nonabelian Gauge Theories*, *Nucl. Phys. B* **291** (1987) 392 [[INSPIRE](#)].
- [2] R. Kleiss and W.J. Stirling, *Spinor Techniques for Calculating  $p\bar{p} \rightarrow W^\pm/Z^0 + \text{Jets}$* , *Nucl. Phys. B* **262** (1985) 235 [[INSPIRE](#)].
- [3] S.J. Parke and T.R. Taylor, *An Amplitude for  $n$  Gluon Scattering*, *Phys. Rev. Lett.* **56** (1986) 2459 [[INSPIRE](#)].
- [4] R. Britto, F. Cachazo and B. Feng, *New recursion relations for tree amplitudes of gluons*, *Nucl. Phys. B* **715** (2005) 499 [[hep-th/0412308](#)] [[INSPIRE](#)].
- [5] R. Britto, F. Cachazo, B. Feng and E. Witten, *Direct proof of tree-level recursion relation in Yang-Mills theory*, *Phys. Rev. Lett.* **94** (2005) 181602 [[hep-th/0501052](#)] [[INSPIRE](#)].
- [6] E. Witten, *Perturbative gauge theory as a string theory in twistor space*, *Commun. Math. Phys.* **252** (2004) 189 [[hep-th/0312171](#)] [[INSPIRE](#)].
- [7] N. Berkovits, *An alternative string theory in twistor space for  $N = 4$  superYang-Mills*, *Phys. Rev. Lett.* **93** (2004) 011601 [[hep-th/0402045](#)] [[INSPIRE](#)].
- [8] R. Roiban, M. Spradlin and A. Volovich, *On the tree level  $S$  matrix of Yang-Mills theory*, *Phys. Rev. D* **70** (2004) 026009 [[hep-th/0403190](#)] [[INSPIRE](#)].
- [9] V.P. Nair, *A Current Algebra for Some Gauge Theory Amplitudes*, *Phys. Lett. B* **214** (1988) 215 [[INSPIRE](#)].
- [10] N. Berkovits and E. Witten, *Conformal supergravity in twistor-string theory*, *JHEP* **08** (2004) 009 [[hep-th/0406051](#)] [[INSPIRE](#)].

- [11] D. Skinner, *Twistor Strings for  $N = 8$  Supergravity*, [arXiv:1301.0868](#) [[INSPIRE](#)].
- [12] F. Cachazo, S. He and E.Y. Yuan, *Scattering equations and Kawai-Lewellen-Tye orthogonality*, *Phys. Rev. D* **90** (2014) 065001 [[arXiv:1306.6575](#)] [[INSPIRE](#)].
- [13] F. Cachazo, S. He and E.Y. Yuan, *Scattering of Massless Particles in Arbitrary Dimensions*, *Phys. Rev. Lett.* **113** (2014) 171601 [[arXiv:1307.2199](#)] [[INSPIRE](#)].
- [14] F. Cachazo, S. He and E.Y. Yuan, *Scattering Equations and Matrices: From Einstein To Yang-Mills, DBI and NLSM*, *JHEP* **07** (2015) 149 [[arXiv:1412.3479](#)] [[INSPIRE](#)].
- [15] D.J. Gross and P.F. Mende, *The High-Energy Behavior of String Scattering Amplitudes*, *Phys. Lett. B* **197** (1987) 129 [[INSPIRE](#)].
- [16] D.B. Fairlie and D.E. Roberts, *Dual models without tachyons — a new approach*, (1972), [[INSPIRE](#)].
- [17] M. Spradlin and A. Volovich, *From Twistor String Theory To Recursion Relations*, *Phys. Rev. D* **80** (2009) 085022 [[arXiv:0909.0229](#)] [[INSPIRE](#)].
- [18] J.A. Farrow and A.E. Lipstein, *From 4d Ambitwistor Strings to On Shell Diagrams and Back*, *JHEP* **07** (2017) 114 [[arXiv:1705.07087](#)] [[INSPIRE](#)].
- [19] D. Nandan, A. Volovich and C. Wen, *A Grassmannian Etude in NMHV Minors*, *JHEP* **07** (2010) 061 [[arXiv:0912.3705](#)] [[INSPIRE](#)].
- [20] N. Arkani-Hamed, J. Bourjaily, F. Cachazo and J. Trnka, *Unification of Residues and Grassmannian Dualities*, *JHEP* **01** (2011) 049 [[arXiv:0912.4912](#)] [[INSPIRE](#)].
- [21] Y. Geyer, A.E. Lipstein and L.J. Mason, *Ambitwistor Strings in Four Dimensions*, *Phys. Rev. Lett.* **113** (2014) 081602 [[arXiv:1404.6219](#)] [[INSPIRE](#)].
- [22] I. Bandos, *Twistor/ambitwistor strings and null-superstrings in spacetime of  $D = 4, 10$  and 11 dimensions*, *JHEP* **09** (2014) 086 [[arXiv:1404.1299](#)] [[INSPIRE](#)].
- [23] J.A. Farrow and A.E. Lipstein, *New Worldsheet Formulae for Conformal Supergravity Amplitudes*, *JHEP* **07** (2018) 074 [[arXiv:1805.04504](#)] [[INSPIRE](#)].
- [24] T. Adamo, E. Casali and D. Skinner, *Ambitwistor strings and the scattering equations at one loop*, *JHEP* **04** (2014) 104 [[arXiv:1312.3828](#)] [[INSPIRE](#)].
- [25] Y. Geyer, L. Mason, R. Monteiro and P. Tourkine, *Loop Integrands for Scattering Amplitudes from the Riemann Sphere*, *Phys. Rev. Lett.* **115** (2015) 121603 [[arXiv:1507.00321](#)] [[INSPIRE](#)].
- [26] F. Cachazo, S. He and E.Y. Yuan, *One-Loop Corrections from Higher Dimensional Tree Amplitudes*, *JHEP* **08** (2016) 008 [[arXiv:1512.05001](#)] [[INSPIRE](#)].
- [27] C. Baadsgaard, N.E.J. Bjerrum-Bohr, J.L. Bourjaily, S. Caron-Huot, P.H. Damgaard and B. Feng, *New Representations of the Perturbative  $S$ -matrix*, *Phys. Rev. Lett.* **116** (2016) 061601 [[arXiv:1509.02169](#)] [[INSPIRE](#)].
- [28] A.E. Lipstein, *Soft Theorems from Conformal Field Theory*, *JHEP* **06** (2015) 166 [[arXiv:1504.01364](#)] [[INSPIRE](#)].
- [29] B.U.W. Schwab and A. Volovich, *Subleading Soft Theorem in Arbitrary Dimensions from Scattering Equations*, *Phys. Rev. Lett.* **113** (2014) 101601 [[arXiv:1404.7749](#)] [[INSPIRE](#)].
- [30] F. Cachazo, S. He and E.Y. Yuan, *New Double Soft Emission Theorems*, *Phys. Rev. D* **92** (2015) 065030 [[arXiv:1503.04816](#)] [[INSPIRE](#)].

- [31] T. Adamo, E. Casali and D. Skinner, *Perturbative gravity at null infinity*, *Class. Quant. Grav.* **31** (2014) 225008 [[arXiv:1405.5122](#)] [[INSPIRE](#)].
- [32] Y. Geyer, A.E. Lipstein and L. Mason, *Ambitwistor strings at null infinity and (subleading) soft limits*, *Class. Quant. Grav.* **32** (2015) 055003 [[arXiv:1406.1462](#)] [[INSPIRE](#)].
- [33] D. Nandan, J. Plefka and W. Wormsbecher, *Collinear limits beyond the leading order from the scattering equations*, *JHEP* **02** (2017) 038 [[arXiv:1608.04730](#)] [[INSPIRE](#)].
- [34] D. Nandan, J. Plefka, O. Schlotterer and C. Wen, *Einstein-Yang-Mills from pure Yang-Mills amplitudes*, *JHEP* **10** (2016) 070 [[arXiv:1607.05701](#)] [[INSPIRE](#)].
- [35] C. Baadsgaard, N.E.J. Bjerrum-Bohr, J.L. Bourjaily and P.H. Damgaard, *Integration Rules for Scattering Equations*, *JHEP* **09** (2015) 129 [[arXiv:1506.06137](#)] [[INSPIRE](#)].
- [36] C. Baadsgaard, N.E.J. Bjerrum-Bohr, J.L. Bourjaily, P.H. Damgaard and B. Feng, *Integration Rules for Loop Scattering Equations*, *JHEP* **11** (2015) 080 [[arXiv:1508.03627](#)] [[INSPIRE](#)].
- [37] L. Dolan and P. Goddard, *The Polynomial Form of the Scattering Equations*, *JHEP* **07** (2014) 029 [[arXiv:1402.7374](#)] [[INSPIRE](#)].
- [38] L.J. Dixon, J.M. Henn, J. Plefka and T. Schuster, *All tree-level amplitudes in massless QCD*, *JHEP* **01** (2011) 035 [[arXiv:1010.3991](#)] [[INSPIRE](#)].
- [39] J.L. Bourjaily, *Efficient Tree-Amplitudes in  $N = 4$ : Automatic BCFW Recursion in Mathematica*, [arXiv:1011.2447](#) [[INSPIRE](#)].
- [40] M. Søgaard and Y. Zhang, *Scattering Equations and Global Duality of Residues*, *Phys. Rev. D* **93** (2016) 105009 [[arXiv:1509.08897](#)] [[INSPIRE](#)].
- [41] H. Elvang and Y.-t. Huang, *Scattering Amplitudes in Gauge Theory and Gravity*, Cambridge University Press, (2015).
- [42] J.M. Henn and J.C. Plefka, *Scattering Amplitudes in Gauge Theories*, *Lect. Notes Phys.* **883** (2014) pp.1 [[INSPIRE](#)].
- [43] H. Elvang, Y.-t. Huang and C. Peng, *On-shell superamplitudes in  $N < 4$  SYM*, *JHEP* **09** (2011) 031 [[arXiv:1102.4843](#)] [[INSPIRE](#)].
- [44] N.J.A. Sloane, *Triangle of Eulerian numbers  $T(n,k)$* , <https://oeis.org/A008292>.
- [45] S. Weinzierl, *On the solutions of the scattering equations*, *JHEP* **04** (2014) 092 [[arXiv:1402.2516](#)] [[INSPIRE](#)].
- [46] V. Alekseev, *Abel's Theorem in Problems and Solutions*, Springer Netherlands, (2004).
- [47] C. Kalousios, *Massless scattering at special kinematics as Jacobi polynomials*, *J. Phys. A* **47** (2014) 215402 [[arXiv:1312.7743](#)] [[INSPIRE](#)].
- [48] G.P. Lepage, *A New Algorithm for Adaptive Multidimensional Integration*, *J. Comput. Phys.* **27** (1978) 192 [[INSPIRE](#)].
- [49] J. Alwall et al., *The automated computation of tree-level and next-to-leading order differential cross sections and their matching to parton shower simulations*, *JHEP* **07** (2014) 079 [[arXiv:1405.0301](#)] [[INSPIRE](#)].
- [50] Y. Geyer and R. Monteiro, *Gluons and gravitons at one loop from ambitwistor strings*, *JHEP* **03** (2018) 068 [[arXiv:1711.09923](#)] [[INSPIRE](#)].

- [51] Y. Geyer and R. Monteiro, *Two-Loop Scattering Amplitudes from Ambitwistor Strings: from Genus Two to the Nodal Riemann Sphere*, [arXiv:1805.05344](#) [[INSPIRE](#)].
- [52] A. Brandhuber, E. Hughes, R. Panerai, B. Spence and G. Travaglini, *The connected prescription for form factors in twistor space*, *JHEP* **11** (2016) 143 [[arXiv:1608.03277](#)] [[INSPIRE](#)].
- [53] S. He and Y. Zhang, *Connected formulas for amplitudes in standard model*, *JHEP* **03** (2017) 093 [[arXiv:1607.02843](#)] [[INSPIRE](#)].
- [54] K.A. Roehrig, *Chiral splitting and  $\mathcal{N} = 4$  Einstein-Yang-Mills tree amplitudes in 4d*, *JHEP* **08** (2017) 033 [[arXiv:1705.09315](#)] [[INSPIRE](#)].